Performance of Triangular Fault Elements in Earthquake Simulators

SCEC Project #14090

Final Report Submitted to SCEC
February 23, 2015

Principal Investigators:

Terry Tullis (Brown University)

Michael Barall (Invisible Software)

Contents

1. Introduction

An earthquake simulator is a computer program that generates a synthetic earthquake catalog spanning thousands of years, or longer. Most of the computational effort in an earthquake simulator goes into computing how slip on one part of a fault affects stresses on other parts of the fault, and on other faults. The computation is done by discretizing the fault system into a large number of fault elements, and using Greens functions to determine how a pattern of slip on some fault elements affects the stresses on all the fault elements (Tullis *et al.* 2012; Ward 2012; Sachs *et al.* 2012; Pollitz 2012; Richards-Dinger and Dieterich 2012).

Traditionally, earthquake simulators have used rectangular fault elements, chosen so that the Okada Greens functions can be used (Okada 1992). Recently, due to the development of new Greens functions for triangular dislocations, it has become practical to use triangular fault elements (Meade 2007; Gimbutas *et al.* 2012). The purpose of this project is to assess the accuracy of stress calculations performed with triangular fault elements, as compared to the accuracy of the same calculations done with rectangular fault elements.

For planar faults, rectangles and triangles can be expected to give the same results (except possibly if the fault has an irregular boundary, which triangles can follow more accurately than rectangles). But for curved faults, rectangles and triangles give different results. When a fault is curved, partitioning it into rectangular fault elements will necessarily create gaps and overlaps between adjacent elements. (It should be noted that the Okada Greens functions do not work with arbitrary rectangles, but instead require rectangles whose upper and lower edges are horizontal. Meeting this constraint further exacerbates the creation of gaps and overlaps.) In contrast, partitioning a curved fault into triangular fault elements can be done using a triangular mesh which has no gaps or overlaps between adjacent elements.

Because triangles can represent curved fault geometry more accurately than rectangles, one intuitively expects that stress calculations performed with triangles should be more accurate than stress calculations done with rectangles. The original motivation for this project was to demonstrate that this intuitive expectation is true, and quantify the difference in accuracy.

However, our results are contrary to the intuitive expectation. In our tests, triangles are not superior to rectangles. One or the other may be superior in a particular case, but, overall, rectangles perform as well as or better than triangles. Another unexpected result is that one triangulation of a fault surface may perform significantly better than another triangulation with a different pattern of triangles.

2. Research Strategy

We performed stress computations on two curved fault surfaces, one with negative curvature and one with positive curvature.

Our negatively-curved surface is motivated by the shape of the San Andreas Fault in southern California (Fuis *et al.* 2012). We ran our tests on a strike-slip fault in the shape of a helicoid, with dip angles ranging from +45 degrees at one end of the fault to −45 degrees at the other end of the fault, passing through 90 degrees at the center of the fault, and with a straight fault trace.

Our positively-curved surface is motivated by the shape of the Cascadia subduction fault (McCrory *et al.* 2012). We ran our tests on a thrust fault in the shape of an ellipsoid, with dip angles ranging from 10 degrees at the earth's surface to 30 degrees at the base of the fault, and strike angles ranging from +30 degrees to −30 degrees.

For both surfaces, we used greater curvature than the natural fault, in order to emphasize the effects of the curvature.

Our strategy is to impose uniform slip on part of the fault surface, and then compute the induced shear and normal stresses elsewhere on the fault surface. We examined the induced stresses in two places: immediately adjacent to the slipping portion of the fault, and 4.8 km away from the slipping portion. The induced stresses immediately adjacent to the slipping area are relevant for the simulation of rupture propagation during an earthquake, because the earthquake rupture proceeds from the currently-slipping elements to their neighboring elements. The stresses 4.8 km away from the slipping area are relevant for examining the accuracy of stress transfer over larger distances, particularly for simulators that employ rate-state friction, in which stress changes influence the evolution of fault state even before the onset of seismic slip. We chose a separation of 4.8 km because it is roughly the maximum size of a fault-to-fault jump (Wesnousky 2006).

To assess the accuracy of the calculation, we discretize the fault surfaces using either rectangles or triangles, each in six different sizes: 300, 600, 1200, 2400, 4800, and 9600 meters. The 300 m results are taken as the reference solution, and the accuracy of the other five sets of results is assessed by how well they approximate the reference value. We now describe in detail how this is done.

Accuracy Assessments

Earthquake simulators use fault elements in two ways: as *sources* and as *targets*. For each pair of fault elements S and T, the simulator computes the shear and normal stress acting on the centroid of the target element T, due to fault slip occurring on the source element S. Each fault element acts as both a source and a target. (Some simulators use both shear and normal stress, while other simulators use only the shear stress.)

When the fault is curved, the fault elements can only approximate the curved surface. When computing stresses, there is some error due to the fact that the source element does not lie precisely in the surface, and there is some error due to the fact that the target element does not lie precisely in the surface. We perform two tests to separately evaluate these two causes of error, and determine whether triangular or rectangular elements are better:

1. A *source test* which evaluates the error originating at the source elements. This test is performed using a source region and target region that are separated by 4800 meters. We vary the discretization of the source region, using either triangles or rectangles with sizes ranging from 600 to 9600 m. The discretization of the target region is held fixed. In this report, the target region is discretized with 600 m rectangles; other discretizations yield similar results. In each case, a reference solution is calculated by discretizing the source region with 300 m elements.

2. A *target test* which evaluates the error originating at the target elements. This test is also performed using a source region and target region that are separated by 4800 meters. The discretization of the source region is held fixed. In this report, the source region is discretized with 300 m rectangles. We vary the discretization of the target region, using either triangles or rectangles with sizes ranging from 600 to 9600 m. In each case, a reference solution is calculated by using 300 m elements in the target region, placed so that their centroids align with the centroids of the larger elements. (That is, the 300 m elements and the larger elements have the same centroids in the (z, d) coordinate system, where z and d represent depth and distance-along-strike as defined below. Note that there will be large gaps between the 300 m elements, because the number of 300 m elements is the same as the number of larger elements.) In order to have a valid comparison, it is necessary to preserve the centroid locations, because simulators evaluate the induced shear and normal stress at the centroids of the target elements.

When an earthquake rupture is propagating along a fault surface, the rupture expands from the region that is currently slipping into the immediately adjacent fault elements. So, it is important for a simulator to be able to accurately compute stresses in the fault elements adjacent to a slipping region. We use a third test to evaluate whether rectangles or triangles are better for this important case:

3. A *propagation test* which evaluates errors occurring during rupture propagation. This test is performed using a target region that borders on the source region, with no separation. We vary the discretization of both the source region and the target region, using either triangles or rectangles with sizes ranging from 600 to 9600 m. The source and target regions are discretized in the same way (*e.g.*, if one is 1200 m triangles then so is the other), as would be the case in a simulator. In each case, a reference solution is calculated by discretizing both the source and target regions with 300 m elements. In the target region, the centroids of the 300 m elements are aligned with the centroids of the larger elements, as in the target test.

Error Metric

A calculation produces one stress value for each target fault element. If there are N target elements, then the stress values can be expressed as an N-dimensional vector $(s_1, \ldots, s_N)$. The components s_i can be either shear stresses or normal stresses. We handle shear stresses and normal stresses separately, so that the vector $(s_1, \ldots, s_N)$ consists of either all shear stresses, or all normal stresses. We also have available a reference solution $(s_1^*, \ldots, s_N^*)$. In order to produce a numerical measure of accuracy, we need to have a metric for comparing $(s_1, \ldots, s_N)$ to $(s_1^*, \ldots, s_N^*)$, that is, we need a metric for comparing N-dimensional vectors.

Our metric is based on the Q metric developed by the SCEC Dynamic Rupture Code Verification Project (Barall and Harris 2015). The Q metric value can range from 0 percent (for perfect agreement) to 200 percent (for vectors pointing in opposite directions); lower values indicate better agreement. Suppose $\boldsymbol{a} = (a_1, \ldots, a_N)$ is an N-dimensional vector. Define the L^1 norm of $\boldsymbol{a}$ to be:

$$\|\boldsymbol{a}\|_1 \equiv \sum_{i=1}^{N} |a_i|$$

Then, given two N-dimensional vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, we compare them using the Q metric defined as:

$$Q \equiv \frac{\|\boldsymbol{a} - \boldsymbol{b}\|_1}{\|\boldsymbol{a}\|_1 + \|\boldsymbol{b}\|_1}$$

In order to express Q as a percentage, the above value is multiplied by 200. All our results use the Q metric, expressed as a percentage, to measure the accuracy of a stress calculation.

3. Fault Geometry and Discretization

We now specify in detail the geometry of our two fault surfaces, the location of the source and target regions, and the manner in which they are partitioned into fault elements.

The fault is embedded in a three-dimensional space with (x, y, z) coordinates. The z coordinate is vertical and points upward, so negative z-values are underground. The average strike of the fault lies parallel to the y-axis.

We introduce a fourth coordinate d, which represents distance along strike (Barall 2012). Every point on the fault surface is identified by the coordinate pair (z, d). In order to specify the three-dimensional shape of the fault, we must specify the mapping $(z, d) \longmapsto (x, y, z)$. In other words, we must specify x and y as functions of z and d.

Negatively-Curved (Helicoidal) Fault Geometry

Our negatively-curved fault surface is a section of a helicoid. It is a strike-slip fault. See Figure 1. Its equations are:

$$x = z \tan(d/r_d)$$

$$y = d$$

$$z_{\min} \leq z \leq z_{\max}$$

$$d_{\min} \leq d \leq d_{\max}$$

The parameters for the helicoid are shown in the following table:

Parameters for Helicoidal Surface	
Parameter	*Value*
r_d	24446.1993 m
$z_{\min}$	−19200 m
$z_{\max}$	0 m
$d_{\min}$	−19200 m
$d_{\max}$	19200 m

With these parameters, the fault trace is a straight line 38400 m long, and the dip angle varies from +45 degrees at one end of the fault to −45 degrees at the other end of the fault.

Remark: We also performed some tests using a helicoid that is 76800 m long, and so has lower curvature. The results were similar to the results shown in this report.
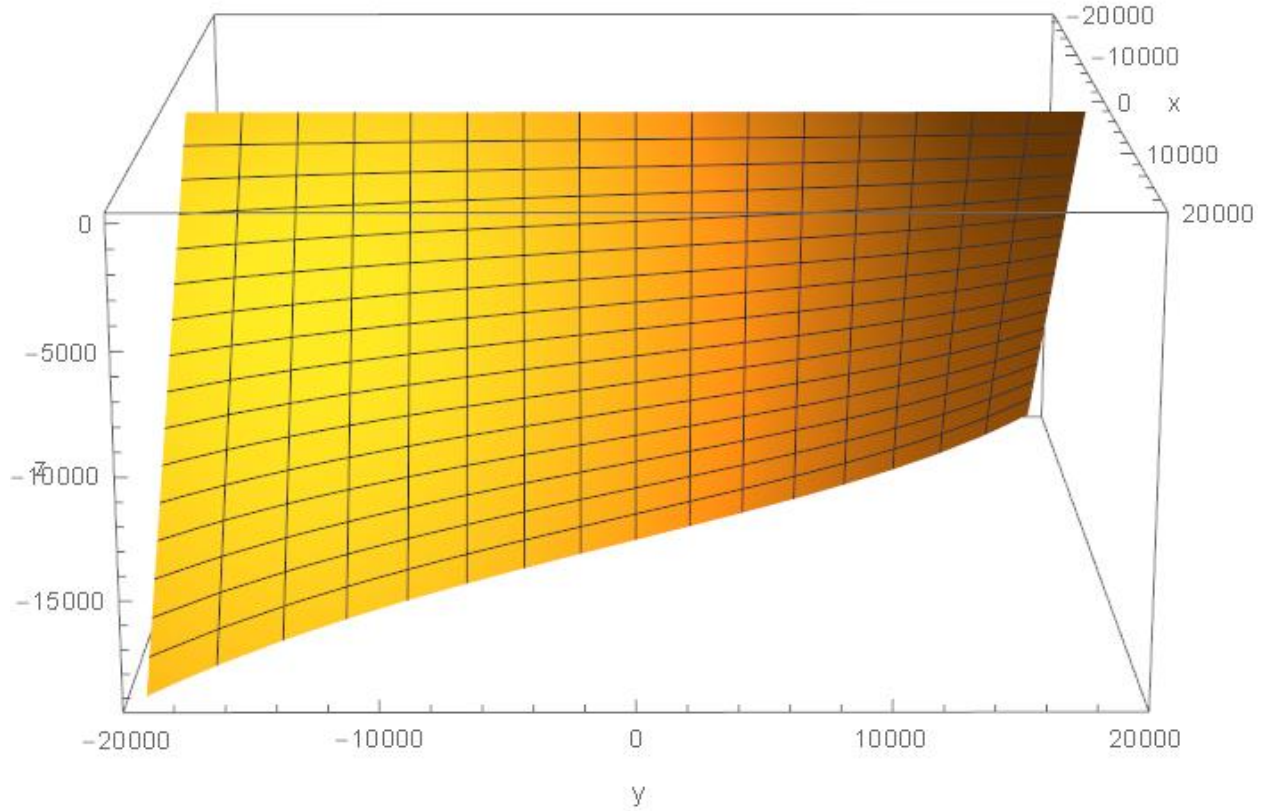
Figure 1. Negatively-curved fault surface. The surface is a section of a helicoid. All dimensions are in meters. The fault trace is a straight line, at $x = 0$, extending from $y = -19200$ to $y = +19200$. Fault dip is -45 degrees at one end of the fault, and $+45$ degrees at the other end of the fault. Dip is 90 degrees in the center of the fault, at $y = 0$. The fault extends from the earth's surface at $z = 0$, to a maximum depth of $z = -19200$. This is a strike-slip fault. Grid lines are contours of constant depth (z) and contours of constant distance-along-strike (d), and do not represent fault elements.

On the negatively-curved fault, we describe tests using four configurations of source region and target region. We designate these four configurations N1 through N4. See Figure 2. The following table defines the four configurations, and shows which tests are performed on each configuration. In the table, all dimensions are in meters, and e denotes the element size.

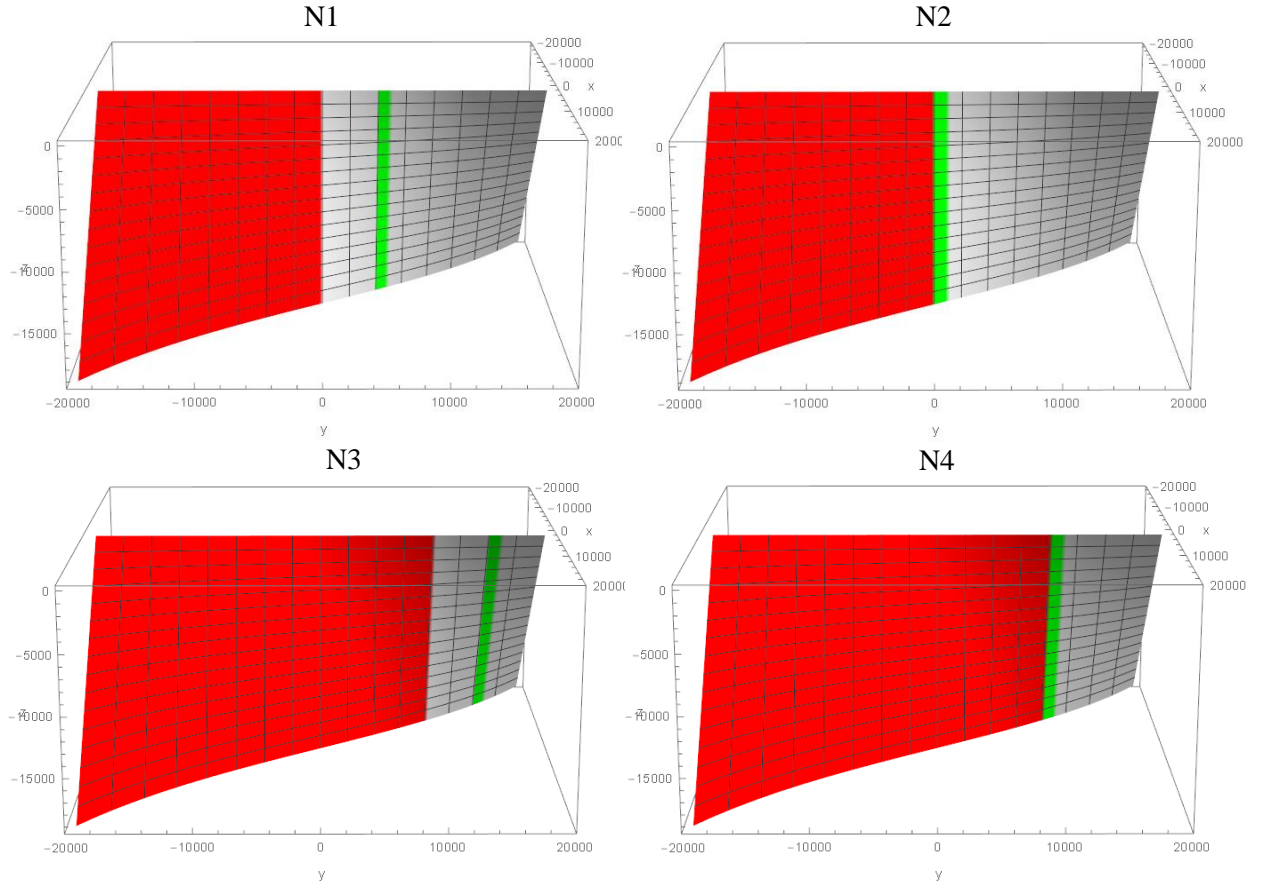| Source and Target Configurations for Helicoidal Surface				
Designation	*Source Region*	*Target Region*	*Tests*	*Direction*
N1	$-19200 \leq z \leq 0$ $-19200 \leq d \leq 0$	$-19200 \leq z \leq 0$ $4800 \leq d \leq 4800 + e$	Source and target tests	Strike-slip
N2	$-19200 \leq z \leq 0$ $-19200 \leq d \leq 0$	$-19200 \leq z \leq 0$ $0 \leq d \leq 0 + e$	Propagation test	Strike-slip
N3	$-19200 \leq z \leq 0$ $-19200 \leq d \leq 9600$	$-19200 \leq z \leq 0$ $14400 \leq d \leq 14400 + e$	Source and target tests	Strike-slip
N4	$-19200 \leq z \leq 0$ $-19200 \leq d \leq 9600$	$-19200 \leq z \leq 0$ $9600 \leq d \leq 9600 + e$	Propagation test	Strike-slip

Figure 2. Source and target configurations for the negatively-curved (helicoidal) fault surface. The four configurations are designated N1 through N4. The source region is shown in red, and the target region is shown in green. The target region is a strip one element thick; the figures assume an element size of 1200 m. Grid lines are contours of constant depth (z) and contours of constant distance-along-strike (d), and do not represent fault elements.
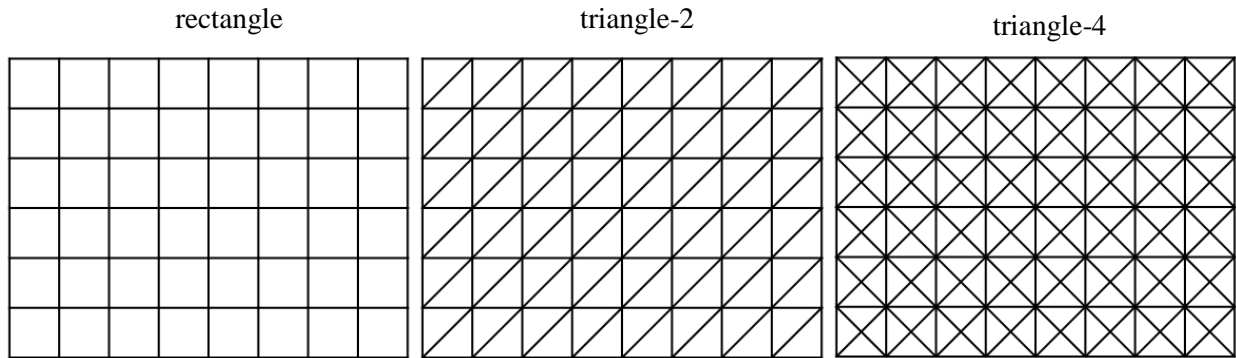
Figure 3. Discretization patterns for the negatively-curved (helicoidal) fault surface. The figure shows three different ways that a portion of the fault surface is partitioned into fault elements. The three methods are "rectangle", "triangle-2", and "triangle-4". See the main text for further description. For the "rectangle" method, the figure shows the fault elements as non-planar quadrilaterals, before they are converted into perfect rectangles. Conversion into perfect rectangles will create gaps and overlaps between adjacent rectangular fault elements. The figure is not to scale.

For each test, we must discretize both the source and target regions. See Figure 3. Discretization is done using either rectangles or triangles, with element sizes of 300, 600, 1200, 2400, 4800, or 9600 meters. To discretize a region with rectangles, we perform the following steps:

1. Working in the (z, d) coordinate system, partition the region into a checkerboard pattern of rectangles. The number of rows and columns in the checkerboard are chosen so that the length and width of each rectangle equals the element size as closely as possible. Due to the fault curvature, this step produces "rectangles" that are actually non-planar quadrilaterals.

2. Modify each rectangle so that its four vertices lie in a plane, and it is of a form suitable for use with the Okada formulas. (See below for procedure.) Due to the fault curvature, this step creates gaps or overlaps between adjacent fault elements.

We explore two different methods for discretizing a region into triangles. We call these methods "triangle-2" and "triangle-4". To discretize a region with triangles, using the triangle-2 method, we perform the following steps:

1. Partition the region into a checkerboard pattern of rectangles, as described above.

2. Cut each rectangle into two triangles, by cutting it along the diagonal that extends from the lower left corner to the upper right corner of the rectangle.

To discretize a region with triangles, using the triangle-4 method, we perform the following steps:

1. Partition the region into a checkerboard pattern of rectangles, as described above.

2. Cut each rectangle into four triangles, by cutting it along both diagonals.

In the test results, we will see that the triangle-4 method yields significantly greater accuracy than the triangle-2 method.

Positively-Curved (Ellipsoidal) Fault Geometry

Our positively-curved fault surface is a section of an ellipsoid. It is a dip-slip fault. See Figure 4. Its equations are:

$$x = x_0 + r_x \cos\left(\frac{d - d_0}{r_d}\right) \sqrt{1 - \left(\frac{z - z_0}{r_z}\right)^2}$$

$$y = y_0 + r_y \sin\left(\frac{d - d_0}{r_d}\right) \sqrt{1 - \left(\frac{z - z_0}{r_z}\right)^2}$$

$$z_{\min} \leq z \leq z_{\max}$$
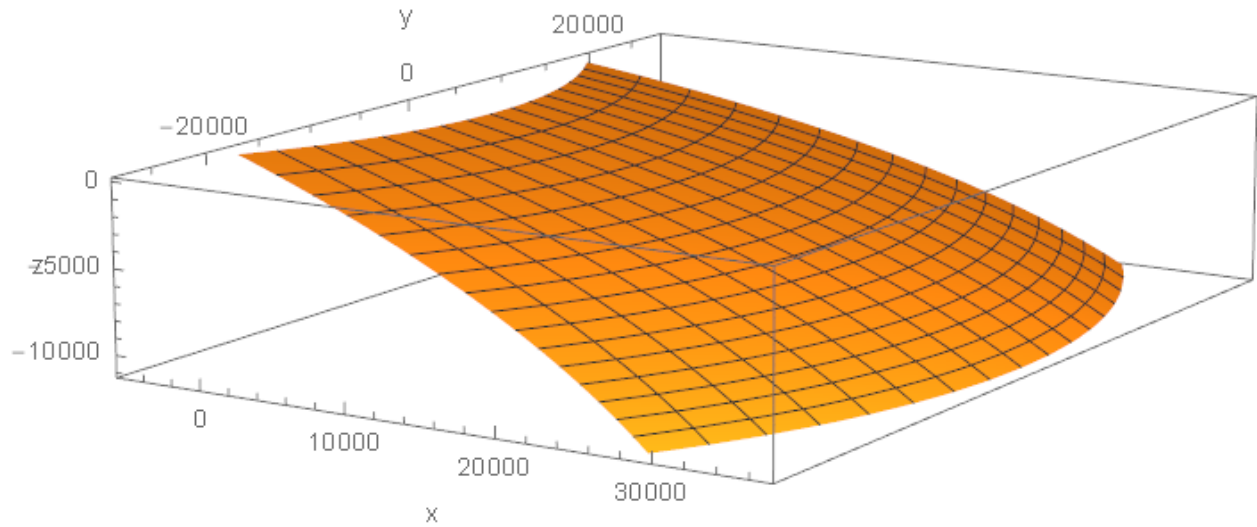
$$d_{\min} \leq d \leq d_{\max}$$

Figure 4. Positively-curved fault surface. The surface is a section of an ellipsoid. All dimensions are in meters. The fault trace is a curve, whose strike angle varies from −30 degrees to +30 degrees. Fault dip is 10 degrees at the top center of the fault, and 30 degrees at the bottom center of the fault. The fault extends from the earth's surface at $z = 0$, to a maximum depth of approximately $z = -11065$. This is a dip-slip (thrust) fault. Grid lines are contours of constant depth (z) and contours of constant distance-along-strike (d), and do not represent fault elements.

The parameters for the ellipsoid are shown in the following table:

Parameters for Ellipsoidal Surface	
Parameter	*Value*
x_0	−73198.5030 m
y_0	0 m
z_0	−20418.7912 m
d_0	0 m
r_x	117619.9820 m
r_y	80203.1670 m
r_z	26085.8149 m
r_d	51192.9964 m
z_{min}	−11064.7197 m
z_{max}	0 m
d_{min}	−19200 m
d_{max}	19200 m

The parameters are chosen to satisfy the following properties, which uniquely determine all the parameters:

- The top edge of the patch, which is the fault trace, has an arc length of 38400 m.

- The bottom edge of the patch has an arc length of 57600 m.

- The curve connecting the center of the top edge to the center of the bottom edge has an arc length of 38400 m.

- At the center of the top edge, the dip angle is 10 degrees.

- At the center of the bottom edge, the dip angle is 30 degrees.

- At the top left and top right corners, which are the ends of the fault trace, the strike angles are +30 degrees and −30 degrees, respectively.
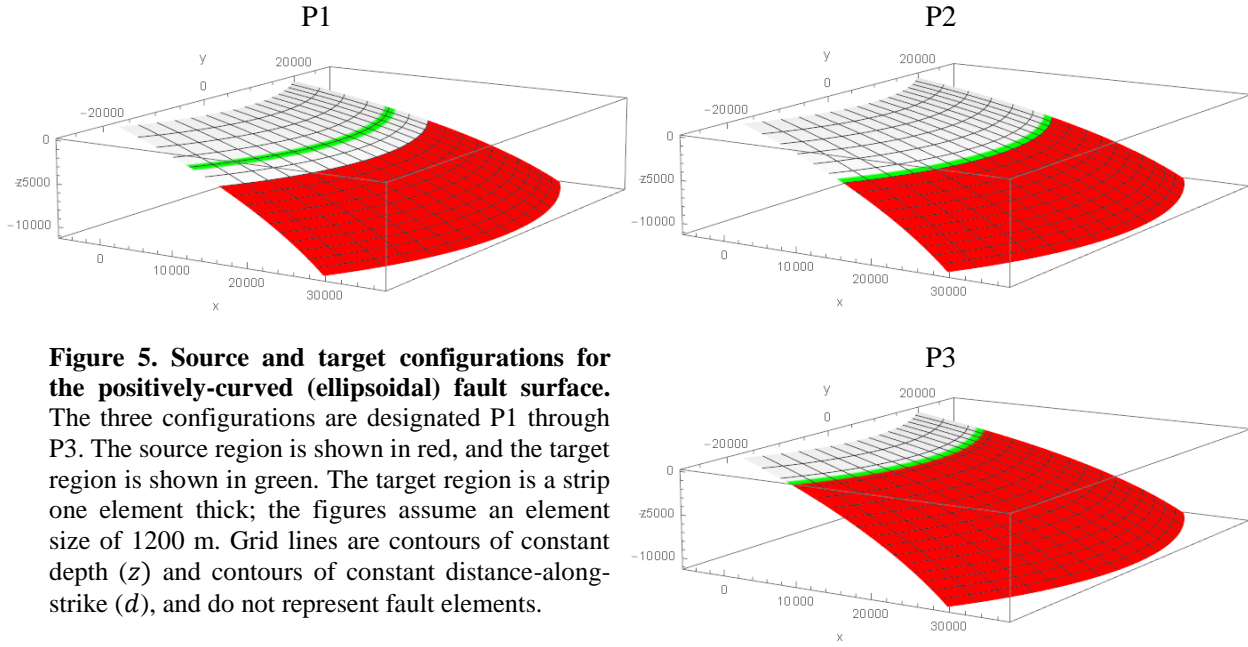
Figure 5. Source and target configurations for the positively-curved (ellipsoidal) fault surface. The three configurations are designated P1 through P3. The source region is shown in red, and the target region is shown in green. The target region is a strip one element thick; the figures assume an element size of 1200 m. Grid lines are contours of constant depth (z) and contours of constant distance-along-strike (d), and do not represent fault elements.

On the positively-curved fault, we describe tests using three configurations of source region and target region. We designate these three configurations P1 through P3. See Figure 5. The following table defines the three configurations, and shows which tests are performed on each configuration. In the table, all dimensions are in meters, and e denotes the element size. Introduce a coordinate v which represents distance down-dip, measured on the fault surface, along an arc from the middle of the fault trace to the middle of the bottom of the fault. The value of v is 0 m at the earth's surface, and −38400 m at the bottom of the fault.

Source and Target Configurations for Ellipsoidal Surface				
Designation	*Source Region*	*Target Region*	*Tests*	*Direction*
P1	$-38400 \leq v \leq -19200$ $-19200 \leq d \leq 19200$	$-14400 \leq v \leq -14400 + e$ $-19200 \leq d \leq 19200$	Source and target tests	Dip-slip
P2	$-38400 \leq v \leq -19200$ $-19200 \leq d \leq 19200$	$-19200 \leq v \leq -19200 + e$ $-19200 \leq d \leq 19200$	Propagation test	Dip-slip
P3	$-38400 \leq v \leq -9600$ $-19200 \leq d \leq 19200$	$-9600 \leq v \leq -9600 + e$ $-19200 \leq d \leq 19200$	Propagation test	Dip-slip

For each test, we must discretize both the source and target regions. See Figure 6. Discretization is done using either rectangles or triangles, with element sizes of 300, 600, 1200, 2400, 4800, or 9600 meters. The discretization procedure is more complicated than for our negatively-curved fault, because our positively-curved fault cannot be partitioned into a checkerboard pattern of rectangles. To discretize a region with rectangles, we perform the following steps:

1. Cut the region into horizontal strips, by slicing it with a set of horizontal planes. The planes are non-uniformly spaced, so that all the strips have equal thickness (*i.e.*, the same change in v) as measured along the fault surface. The number of planes is chosen so that the strip thickness is equal to the element size as closely as possible.

2. For each strip, determine the number of elements that fit along the strip. Divide the length of the strip, as measured along the fault surface, by the element size, and round to the nearest integer.

3. For each strip, construct a horizontal strip of rectangles, with the calculated number of elements. Due to the fault curvature, this step produces "rectangles" that are actually non-planar quadrilaterals.

4. Modify each rectangle so that its four vertices lie in a plane, and it is of a form suitable for use with the Okada formulas. (See below for procedure.) Due to the fault curvature, this step creates gaps or overlaps between adjacent fault elements.
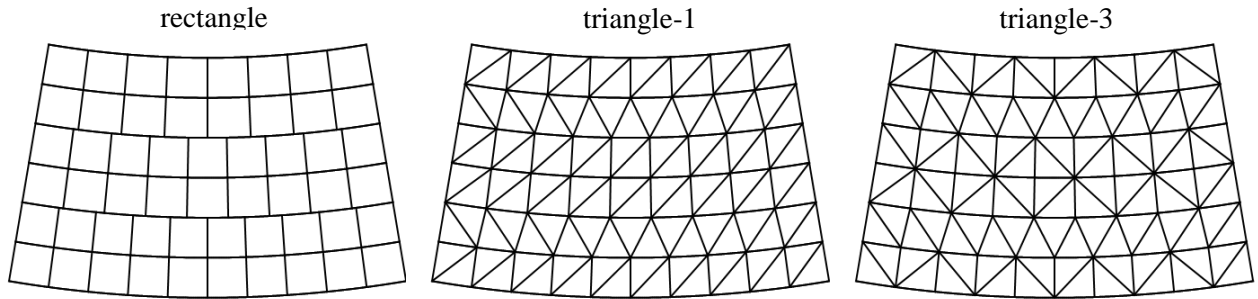
rectangle   triangle-1   triangle-3

Figure 6. Discretization patterns for the positively-curved (ellipsoidal) fault surface. The figure shows three different ways that a portion of the fault surface is partitioned into fault elements. The three methods are "rectangle", "triangle-1", and "triangle-3". See the main text for further description. For the "rectangle" method, the figure shows the fault elements as non-planar quadrilaterals, before they are converted into perfect rectangles. Conversion into perfect rectangles will create gaps and overlaps between adjacent rectangular fault elements. Note that unlike on our negatively-curved surface, the "rectangle" method does not produce a checkerboard pattern. The figure is not to scale.

Notice that because each strip has a different length and a different number of rectangles, the rectangles do not produce a checkboard pattern. In other words, a vertex of a rectangle can lie in the middle of an edge of a rectangle in an adjacent strip. So, unlike our negatively-curved fault, it is not possible to triangulate the surface by starting with rectangles and then cutting the rectangles into triangles.

We explore two different methods for discretizing a region into triangles. We call these methods "triangle-1" and "triangle-3". To discretize a region with triangles we perform the following steps:

1. Cut the region into horizontal strips, as described above.

2. For the top and bottom edge of each strip, determine the number of elements that fit along the edge. Divide the arc length of the edge, as measured along the fault surface, by the element size, and round to the nearest integer.

3. For the top and bottom edge of each strip, lay out equally-spaced nodes along the edge. The number of nodes is one more than the number of elements that fit along the edge.

4. For each strip, construct a strip of triangles that connects the nodes at the top edge to the nodes at the bottom edge.

5. When a strip has the same number of nodes along the top and bottom edges, there is more than one way to construct the strip of triangles. For the "triangle-1" method, the longest triangle edges are all positively sloped (that is, run from bottom-left to top-right). For the "triangle-3" method, the longest triangle edges alternate between positive slope and negative slope.

In the test results, we will see that the triangle-1 method and the triangle-3 method yield almost identical accuracy.

Modification of Rectangular Fault Elements

As noted above, in order to discretize a curved fault surface with rectangles, it is necessary to modify the original non-planar quadrilaterals to form rectangles that are suitable for use with the Okada formulas. We are not aware of a published standard algorithm for performing this modification, although it seems probable that modelers have created such algorithms for their own use. So, we created an algorithm, which we describe here.

Starting with a non-planar quadrilateral, whose four vertices lie in the fault surface, we perform the following steps to convert it into a suitable rectangle:

1. Identify the top edge of the quadrilateral. Calculate the (x, y, z) coordinates of the midpoints of the four edges. The edge whose midpoint has the maximum z coordinate is taken to be the top edge.

2. Calculate the center of the quadrilateral, which is defined to be the average (x, y, z) coordinates of the four vertices.

3. Calculate the strike vector, which is defined to be the midpoint of the right edge minus the midpoint of the left edge.

4. Calculate the dip vector, which is defined to be the midpoint of the top edge minus the midpoint of the bottom edge.

5. Modify the strike vector so that it is horizontal. Do this by setting its z-component equal to zero.

6. Modify the dip vector so that it is perpendicular to the modified strike vector. Do this by adding a multiple of the modified strike vector to the dip vector, choosing the multiple so that the sum is perpendicular to the modified strike vector.

7. The final rectangle is the unique rectangle whose center coincides with the center of the original quadrilateral, and which has the modified strike and dip vectors.

4. Test Results

The following graphs (Figures 7-16) show our test results. The graphs show how the accuracy of the stress calculation varies as the size of the fault elements is varied. There are separate graphs for shear stress and normal stress.

Each graph is a log-log plot, in which the approximation error (Q metric), in percent, is plotted against the number of fault elements. For the source and propagation tests, it is the number of elements in the source region. For the target test, it is the square of the number of elements in the target region, to account for the fact that the size of the target region varies with the element size. (On a logarithmic plot, it makes little difference if the number of elements is squared or not, because squaring just changes the horizontal scale by a factor of two.)

The number of fault elements is the right thing to use on the horizontal axis, because the number of fault elements is what determines the computational effort that a simulator must exert to calculate the stress transfer.

There are three curves on each plot: a blue curve that shows results for rectangular fault elements, and red and green curves that show results for triangular fault elements. The red and green curves correspond to two different methods for constructing the triangulation.

Normal stress always has much larger errors than shear stress. This is due to the fact that for source and target elements on the same fault, normal stresses tend to be smaller than shear stresses, so a small absolute error in normal stress can equate to a large percentage error. Bear in mind that when source and target elements are co-planar, the normal stress is zero except for free-surface effects (and exactly zero if the elements are vertical). So normal stress percentage errors are particularly large for the propagation test, where the source and target elements are adjacent and so are nearly co-planar.
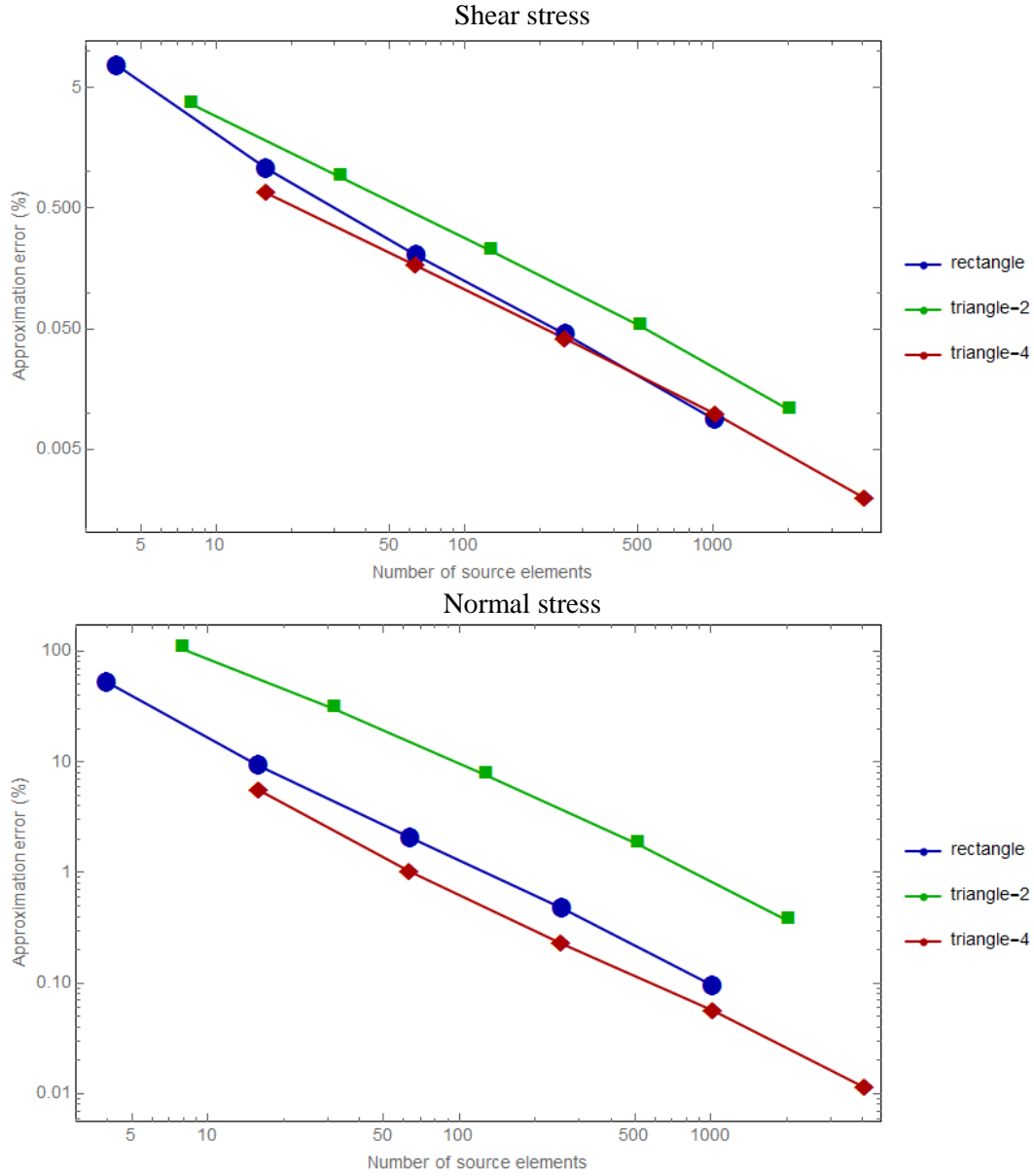
Figure 7. Source Test: Negatively Curved Fault Configuration N1. For shear stress, rectangle and triangle-4 perform about equally well, and both are better than triangle-2. For normal stress, triangle-4 is a little better than rectangle, which in turn is much better than triangle-2.
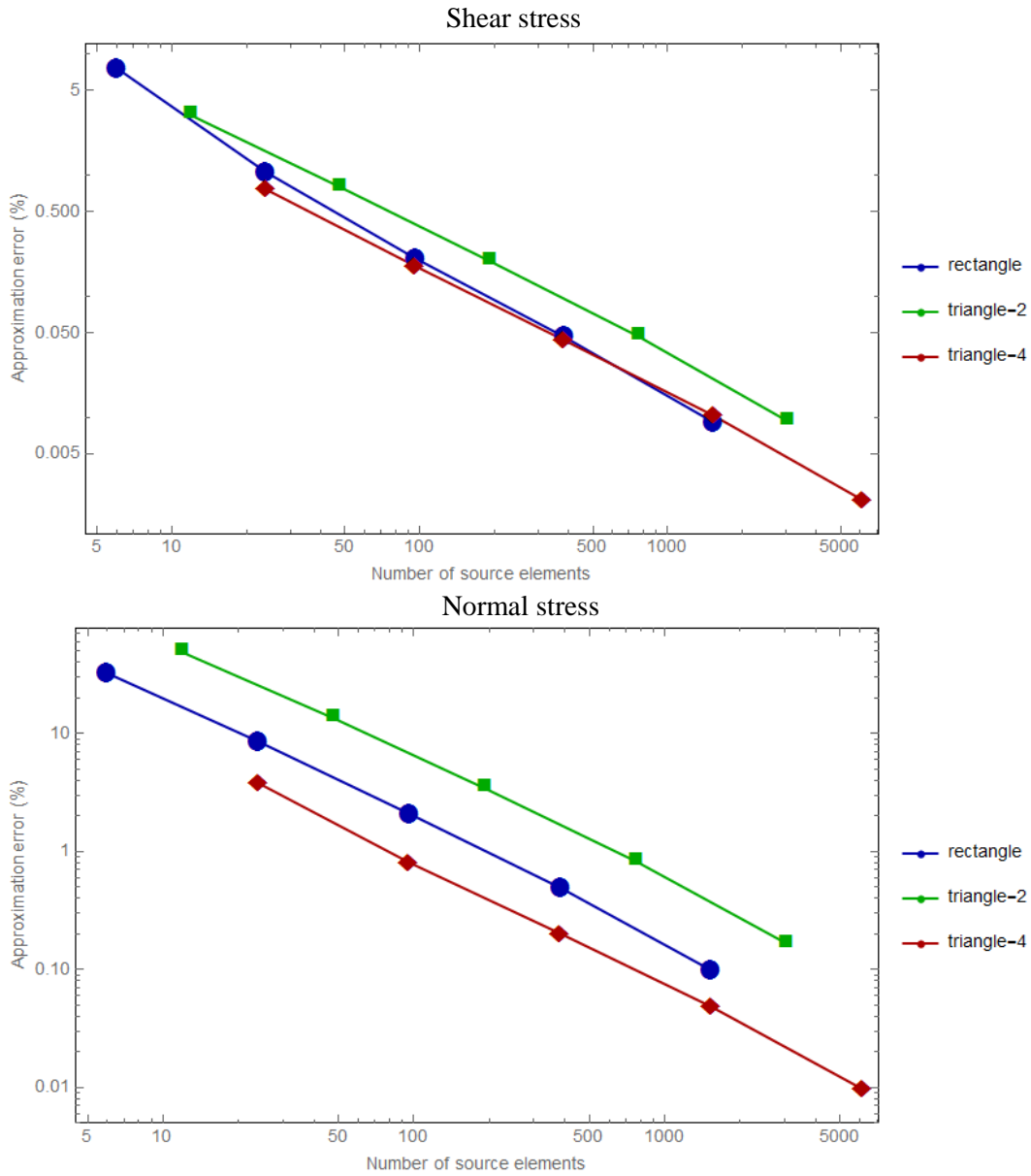
Shear stress

Normal stress

Figure 8. Source Test: Negatively Curved Fault Configuration N3. For shear stress, rectangle and triangle-4 perform about equally well, and both are better than triangle-2. For normal stress, triangle-4 is better than rectangle, which in turn is better than triangle-2. These results are similar to the source test for configuration N1.
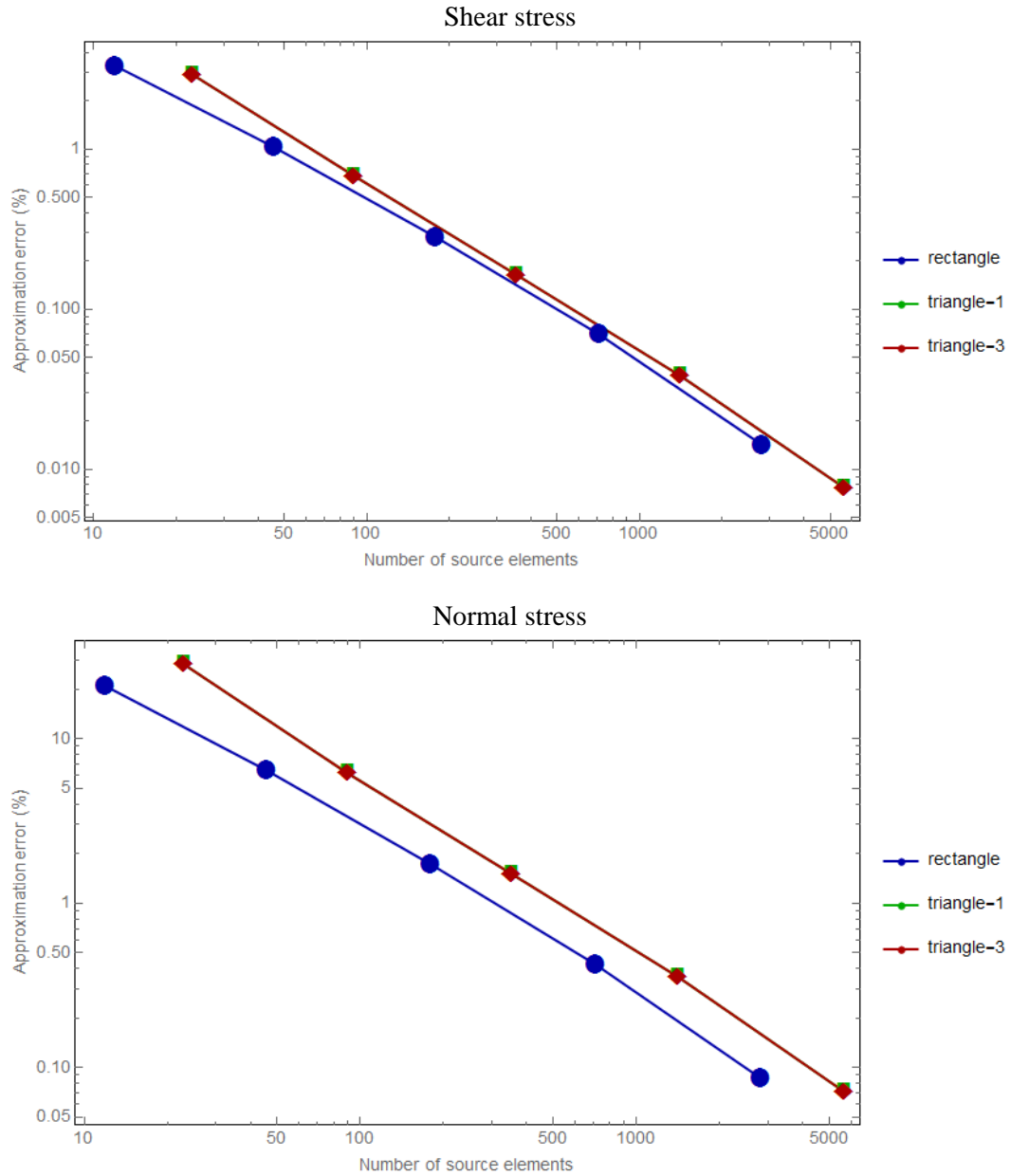
Figure 9. Source Test: Positively Curved Fault Configuration P1. For shear stress, rectangle is slightly better than triangle-1 or triangle-3. For normal stress, rectangle is noticeably better than triangle-1 or triangle-3. The graphs for triangle-1 are difficult to see because they lie almost exactly underneath the graphs for triangle-3.
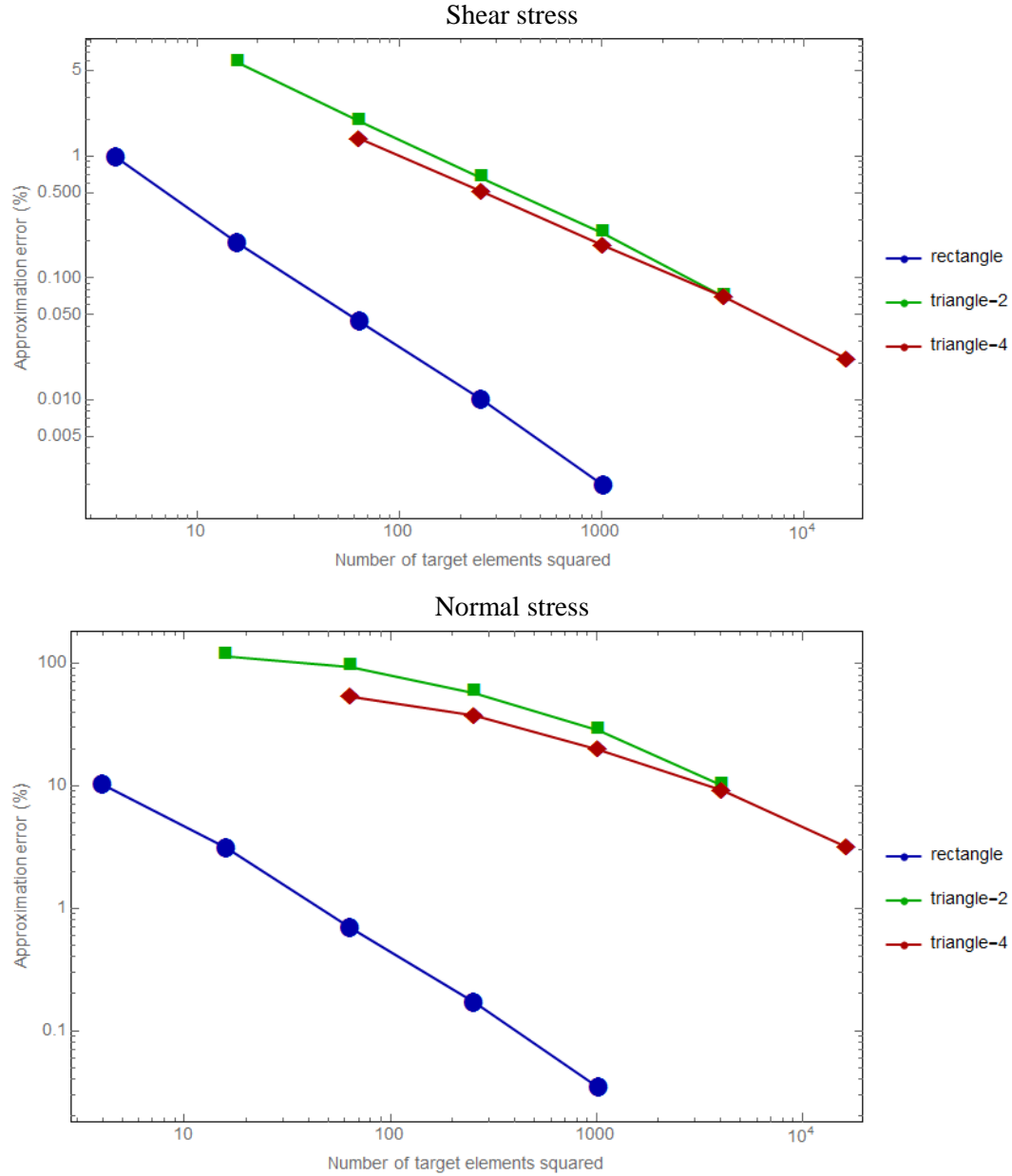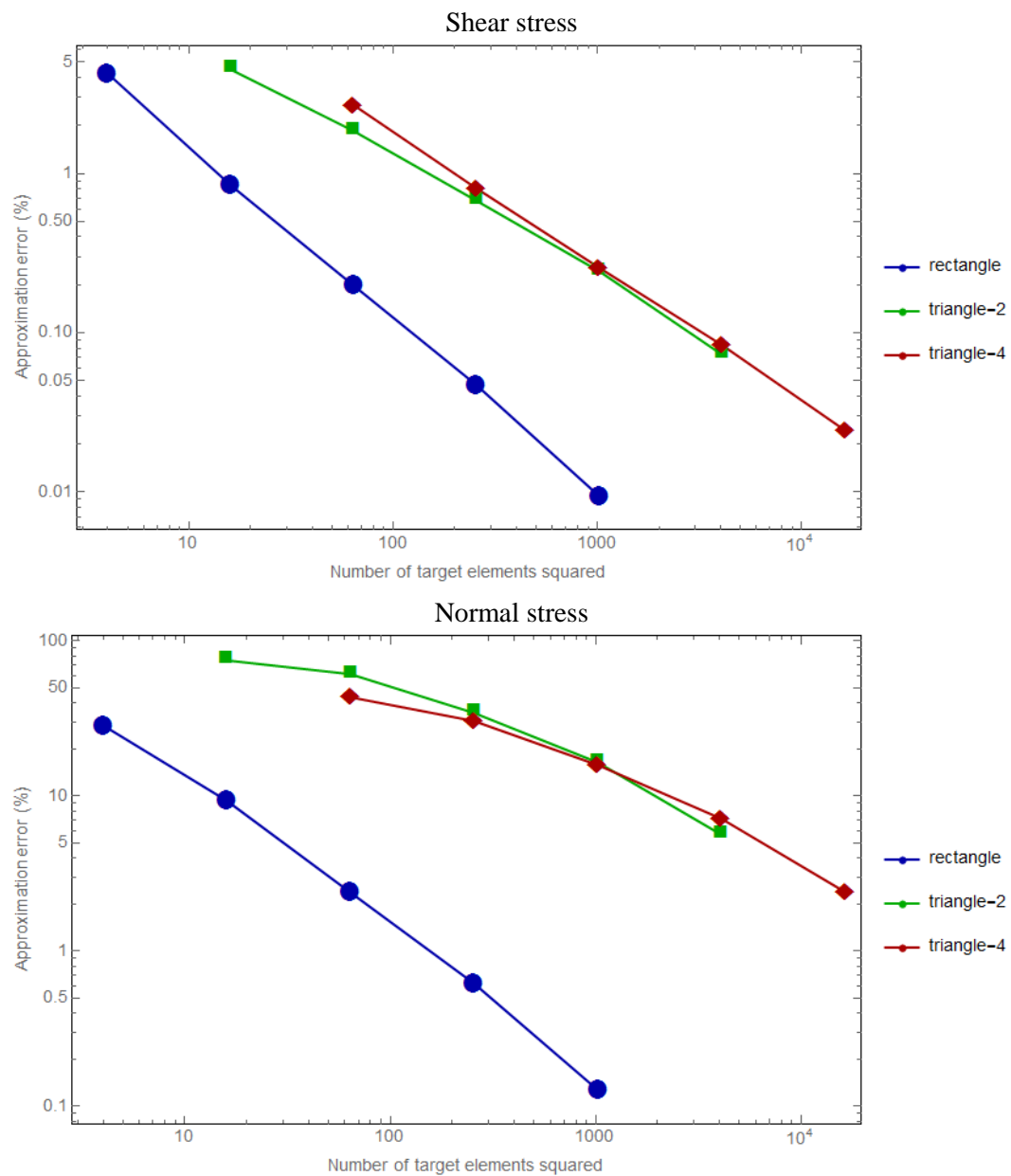
Figure 10. Target Test: Negatively Curved Fault Configuration N1. Rectangle greatly out-performs both triangle-2 and triangle-4, on both shear and normal stress. We attribute the poor performance of triangles, particularly on normal stress, to the fact that on a negatively-curved surface the triangulation is slightly corrugated. That is, differently-oriented triangles have somewhat different strike and dip angles, which generate errors when the stress tensor is resolved onto the triangular fault elements. In contrast, our algorithm for producing rectangular fault elements attempts to preserve the strike and dip angles, which leads to improved accuracy in resolving the stress tensor.

Figure 11. Target Test: Negatively Curved Fault Configuration N3. Rectangle greatly outperforms both triangle-2 and triangle-4, on both shear and normal stress. These results are similar to the target test for configuration N1.
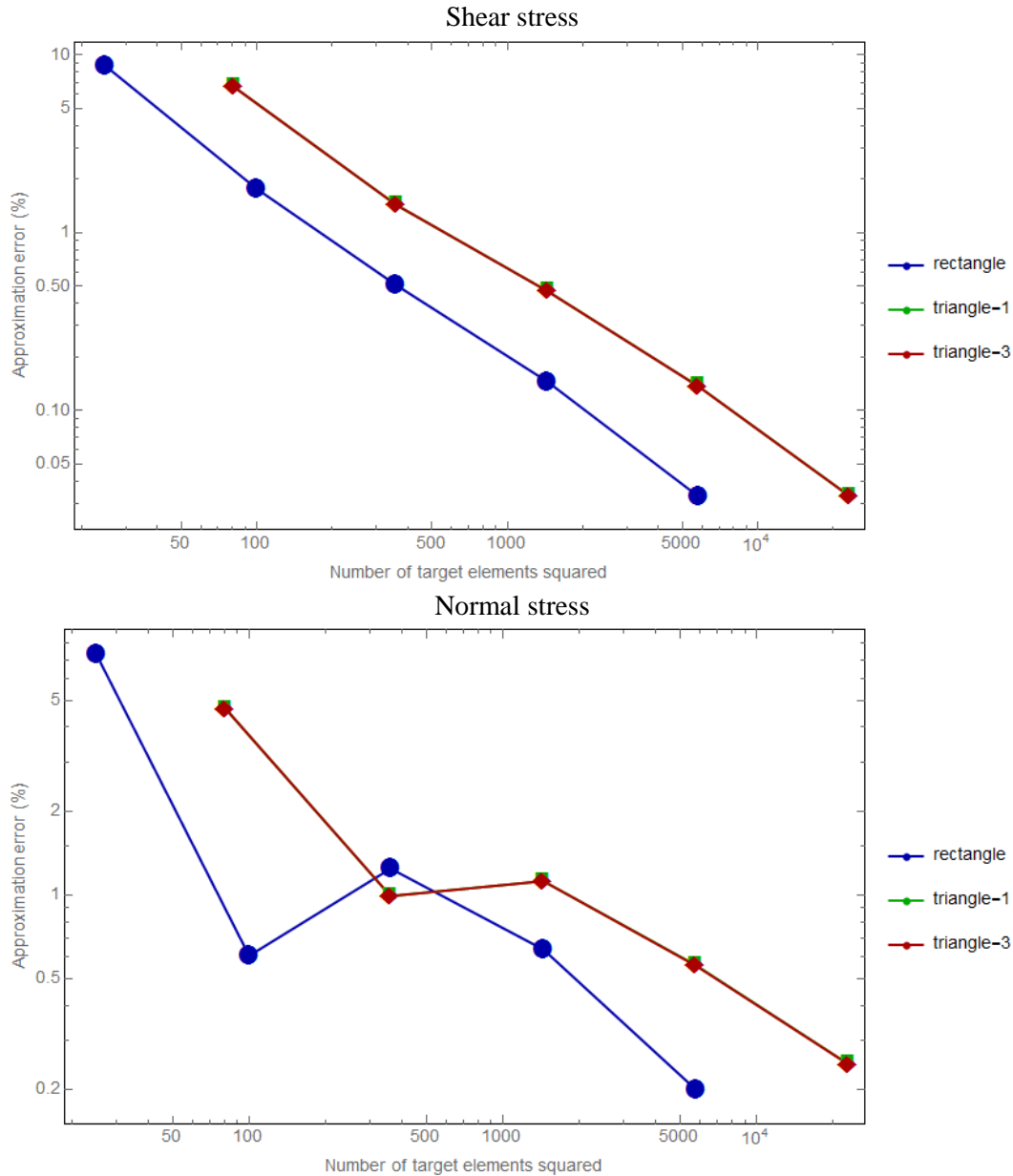
Figure 12. Target Test: Positively Curved Fault Configuration P1. Rectangle out-performs both triangle-1 and triangle-3, for both shear and normal stress, except for one anomalous point on the normal stress plot. Triangles perform much better on the positively-curved fault surface than they did on the negatively-curved surface (although still not as well as rectangles). We attribute this to the fact that the positively-curved fault surface has less tendency to produce corrugations in the triangulation. The graphs for triangle-1 are difficult to see because they lie almost exactly underneath the graphs for triangle-3.
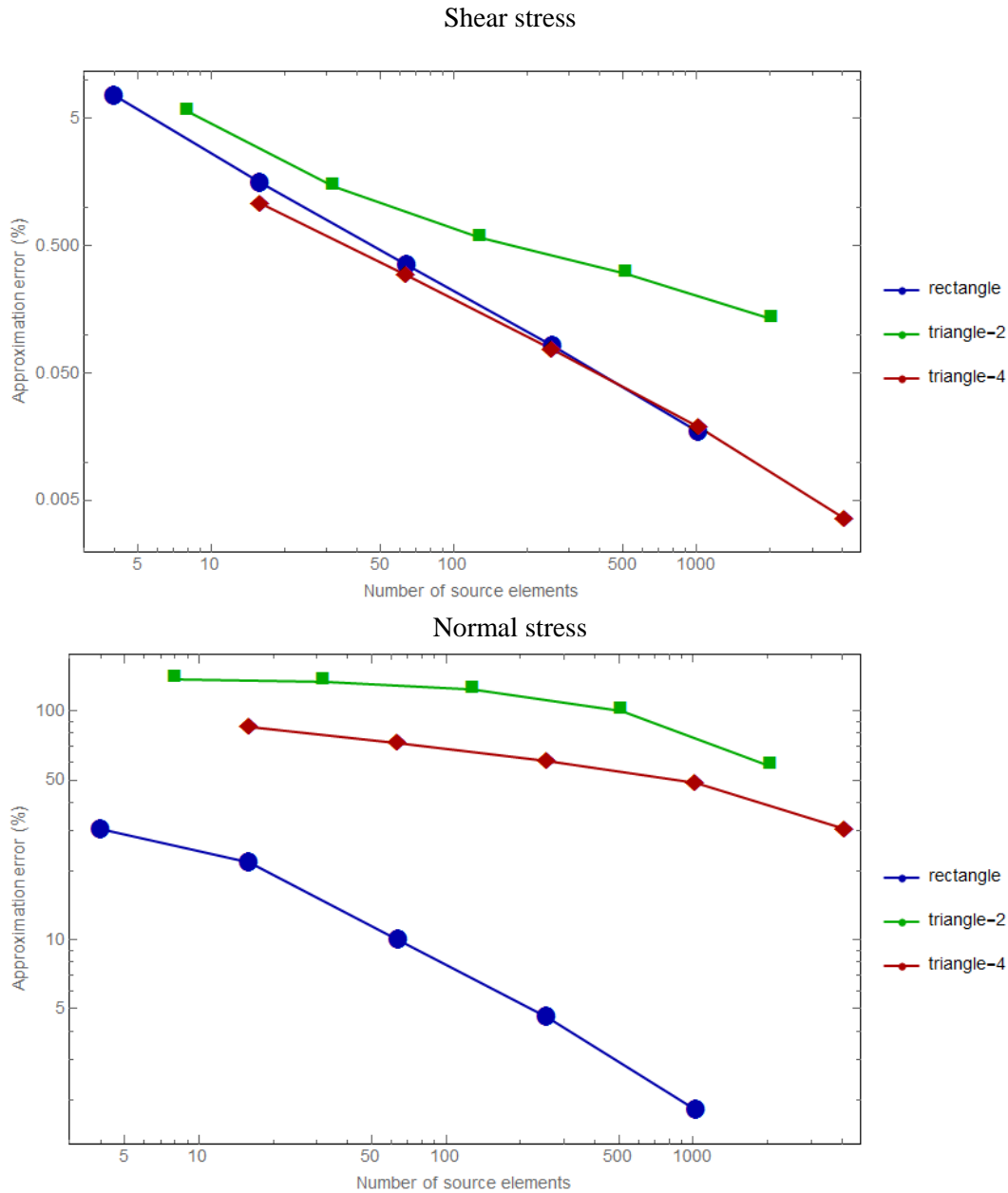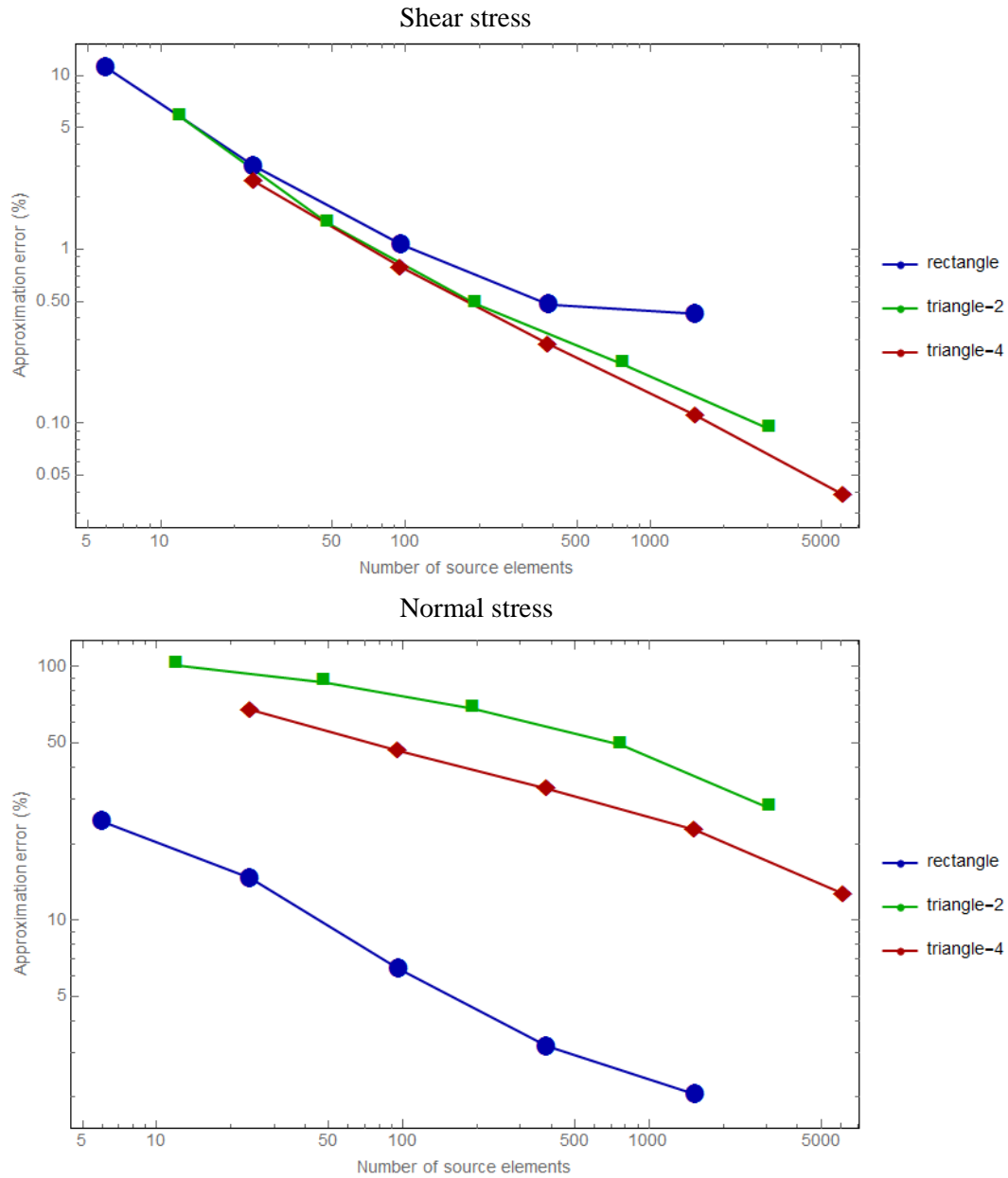
Figure 13. Propagation Test: Negatively Curved Fault Configuration N2. For shear stress, rectangle and triangle-4 perform about equally well, and both are better than triangle-2. For normal stress, rectangle performs much better than triangle-2 or triangle-4. We attribute the large errors in normal stress for triangular elements to the "corrugation" that occurs when triangulating the negatively-curved surface. Bear in mind that in the propagation test, the target elements are almost co-planar with the adjacent source elements, so that small variations in strike and dip angles can produce large percentage errors in the normal stress.

Figure 14. Propagation Test: Negatively Curved Fault Configuration N4. For shear stress, triangle-2 and triangle-4 both perform better than rectangle. Also, there is a hint that the accuracy of rectangle may be leveling off at about 0.5 percent for the smallest elements tested. For normal stress, rectangle performs much better than triangle-2 or triangle-4, in a similar manner to configuration N2.
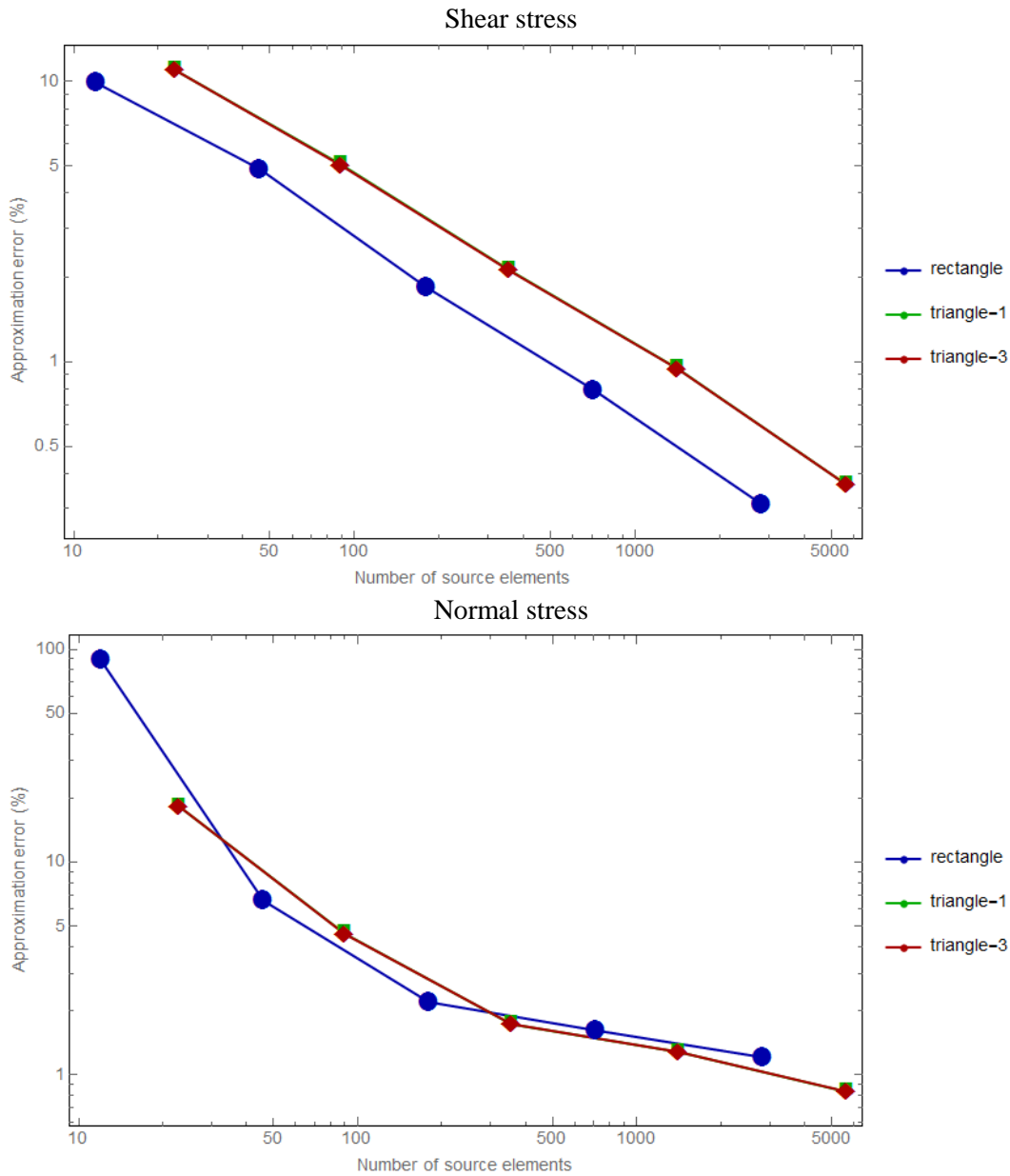
Figure 15. Propagation Test: Positively Curved Fault Configuration P2. For shear stress, rectangle out-performs both triangle-1 and triangle-3. For normal stress, rectangle, triangle-1, and triangle-3 all perform about equally well. The accuracy seems to be leveling off at about 1 percent. One possible explanation for why accuracy might level off is that as elements get smaller, they follow the surface more accurately (which tends to decrease error), but also their centroids are closer together (which tends to increase error). The normal stress results are very different than for the negatively-curved surface, where the triangular elements performed very poorly. The graphs for triangle-1 are difficult to see because they lie almost exactly underneath the graphs for triangle-3.
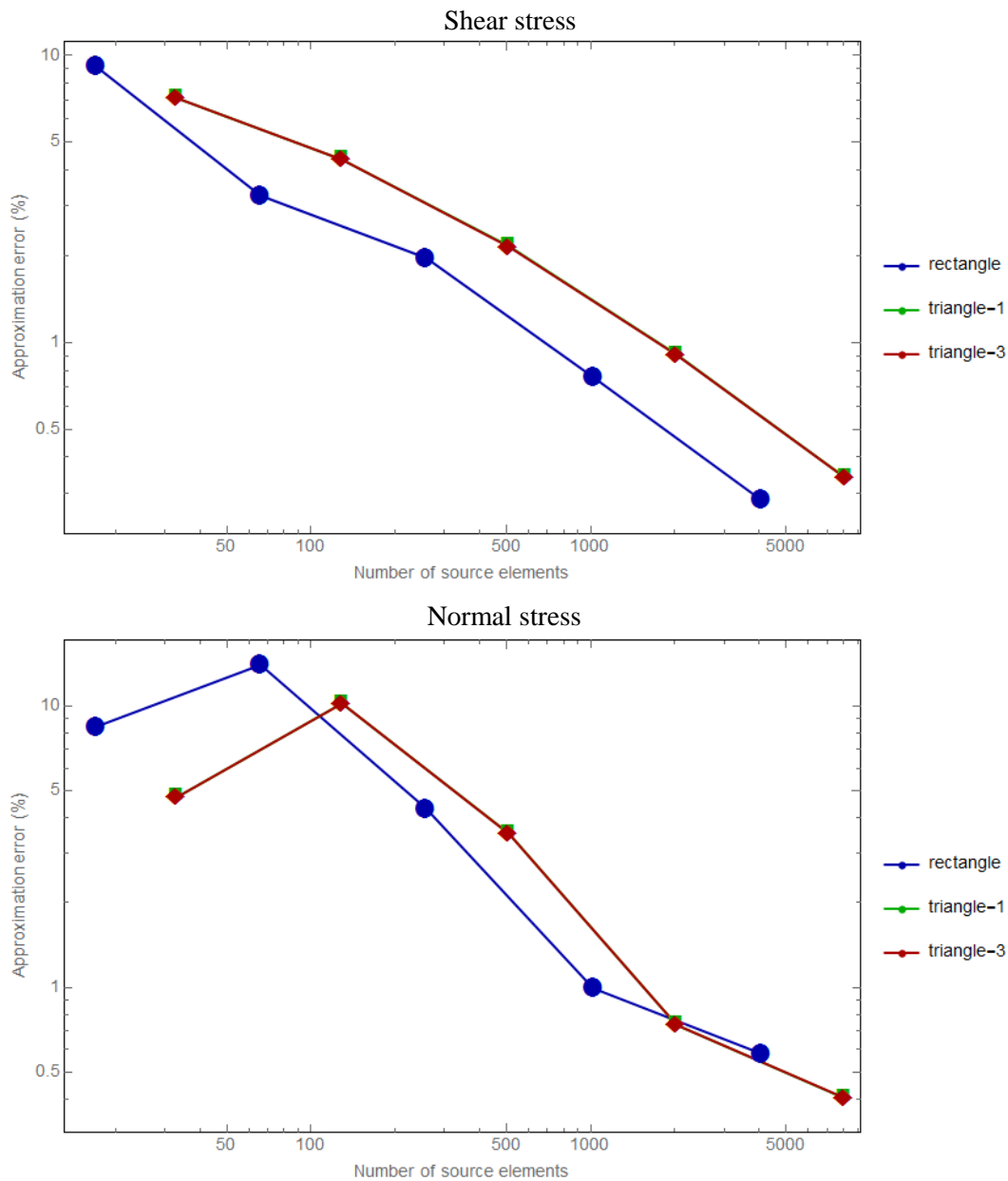
Figure 16. Propagation Test: Positively Curved Fault Configuration P3. For shear stress, rectangle out-performs both triangle-1 and triangle-3. For normal stress, rectangle, triangle-1, and triangle-3 overall perform about equally well, although the rectangles are better in some ranges and the triangles are better in other ranges. There is a hint of leveling-off for the smallest element size tested. Again, we see normal stress results that are very different than for the negatively-curved surface, where triangles performed very poorly. The graphs for triangle-1 are difficult to see because they lie almost exactly underneath the graphs for triangle-3.

5. Conclusions

We performed accuracy tests using both rectangular and triangular fault elements, on two curved fault surfaces: a negatively-curved fault in the shape of a helicoid, and a positively-curved fault in the shape of an ellipsoid. We performed three kinds of tests: a source test that measures the errors when a fault element acts as a dislocation source, a target test that measured the errors when a fault element acts as a target, and a propagation test that measures the errors in propagating a rupture from a slipping region into the immediately adjacent fault elements.

In addition to comparing the performance of rectangles to triangles, we also compared the performance of two different triangulations.

On the positively-curved fault, rectangles performed as well as or better than triangles in every test. In comparing the two triangulations, denoted triangle-1 and triangle-3, we found that the two triangulations performed equally well.

On the negatively-curved fault, results were mixed. In comparing the two triangulations, denoted triangle-2 and triangle-4, we found that triangle-4 performed better than triangle-2 in every test. In the source test, rectangles performed as well as triangle-4 when calculating shear stress. When calculating normal stress, rectangles were less accurate than triangle-4, but still more accurate than triangle-2. In the target test, rectangles were far more accurate than triangles, for both shear and normal stress. In the propagation test, when calculating shear stress, rectangles achieved the same accuracy as triangle-4 in one test, but were less accurate than triangle-2 in the other test. When calculating normal stress, rectangles were far more accurate than triangles.

The poor performance of triangle-2 is noteworthy, because it is the simplest method of constructing a triangulation and so it is frequently used.

Our overall conclusion is that in an earthquake simulator, rectangles should perform as well as triangles.

6. References

Barall, M. (2012), Data Transfer File Formats for Earthquake Simulators, *Seismological Research Letters*, v. 83, n. 6, p. 991-993, doi:10.1785/0220120040.

Barall, M. and R.A. Harris (2015), Metrics for Comparing Dynamic Earthquake Rupture Simulations, *Seismological Research Letters*, v. 86, n. 1, p. 223-235, doi:10.1785/0220140122.

Fuis, G.S., D.S. Scheirer, V.E. Langenheim, and M.D. Kohler (2012), A New Perspective on the Geometry of the San Andreas Fault in Southern California and Its Relationship to Lithospheric Structure, *Bulletin of the Seismological Society of America*, v. 102, n. 1, p. 236-251, doi:10.1785/0120110041.

Gimbutas, Z., L. Greengard, M. Barall, and T.E. Tullis (2012), On the Calculation of Displacement, Stress, and Strain Induced by Triangular Dislocations, *Bulletin of the Seismological Society of America*, v. 102, n. 6, p. 2776-2780, doi:10.1785/0120120127.

McCrory, P.A., J.L. Blair, F. Waldhauser, and D.H. Oppenheimer (2012), Juan de Fuca slab geometry and its relation to Wadati-Benioff zone seismicity, *Journal of Geophysical Research*, v. 117, B09306, doi:10.1029/2012JB009407.

Meade, B.J. (2007), Algorithms for the calculation of exact displacements, strains, and stresses for triangular dislocation elements in a uniform elastic half space, *Computers & Geosciences*, v. 33, n. 8, p. 1064-1075, doi: 10.1016/j.cageo.2006.12.003.

Okada, Y. (1992), Internal deformation due to shear and tensile faults in a half-space, *Bulletin of the Seismological Society of America*, v. 82, n. 2, p. 1018-1040.

Pollitz, F.F. (2012), ViscoSim Earthquake Simulator, *Seismological Research Letters*, v. 83, n. 6, p. 979-982, doi:10.1785/0220120050.

Richards-Dinger, K., and J.H. Dieterich (2012), RSQSim Earthquake Simulator, *Seismological Research Letters*, v. 83, n. 6, p. 983-990, doi:10.1785/0220120105.

Sachs, M.K., E.M. Heien, D.L. Turcotte, M.B. Yikilmaz, J.B. Rundle, and L.H. Kellogg (2012), Virtual California Earthquake Simulator, *Seismological Research Letters*, v. 83, n. 6, p. 973-978, doi:10.1785/0220120052.

Tullis,, T.E., K. Richards-Dinger, M. Barall, J.H. Dieterich, E.H. Field, E.M. Heien, L.H. Kellogg, F.F. Pollitz, J.B. Rundle, M.K. Sachs, D.L. Turcotte, S.N. Ward, and M.B. Yikilmaz (2012), Generic Earthquake Simulator, *Seismological Research Letters*, v. 83, n. 6, p. 959-963, doi:10.1785/0220120093.

Ward, S.N. (2012), ALLCAL Earthquake Simulator, *Seismological Research Letters*, v. 83, n. 6, p. 964-972, doi:10.1785/0220120056

Wesnousky, S.G. (2006), Predicting the endpoints of earthquake ruptures, *Nature*, v. 444, n. 7117, p. 358-360, doi: 10.1038/nature05275.

Invisible Software, Inc.
2058 Coastland Ave.
San Jose, CA 95125
408-265-7436
FAX 408-265-9070

From: Michael Barall

Email: mbinv@invisiblesoft.com

April 7, 2015

John K. McRaney
Southern California Earthquake Center, ZHS 169
University of Southern California
3651 Trousdale Parkway
Los Angeles, CA 90089-0742

Hi John,

This report summarizes the work I have done on SCEC-funded projects for the 2014-2015 fiscal year. I worked on two projects this year:

SCEC Project #14111 "A Collaborative Project: Rupture Dynamics, Validation of the Numerical Simulation Method."

SCEC Project #14090 "Performance of Triangular Fault Elements in Earthquake Simulators."

Feel free to contact me if you have any questions or want additional information.

Michael

Article Published This Year

I published the following article, which describes my work in creating quantitative comparison metrics for the SCEC Dynamic Rupture Code Verification Group:

Barall, M., and R.A. Harris (2015), Metrics for Comparing Dynamic Earthquake Rupture Simulations, *Seismological Research Letters*, vol. 86, no. 1, pages 223-235, doi: 10.1785/0220140122.

The article is available online at: http://srl.geoscienceworld.org/content/86/1/223.full.

Rough Fault with Viscoplasticity Benchmarks

The SCEC Dynamic Rupture Code Verification Group created two benchmarks to test the performance of the dynamic rupture codes in modeling a fault with stochastic roughness:

- TPV29 — Vertical rough fault in an elastic medium.
- TPV30 — Vertical rough fault in a viscoplastic medium.

Both benchmarks use a right-lateral strike-slip fault with stochastic roughness. Figure 1 shows the geometry and the material properties. Figure 2 shows the fault roughness, and the random process that is used to generate it. Measuring 40 km by 20 km, it is the largest fault in any of our benchmarks. The fault needs to be large for two reasons: because viscoplastic yielding builds up gradually as the rupture propagates, and because ample space is needed to let the effects of fault roughness play out. The model also includes gravitational loading, fluid pressure, and a depth-dependent initial stress tensor specified throughout the model volume. The stress tensor is chosen so that the mean fault plane is the optimum plane for fracture (has the highest ratio of shear stress to normal stress). Considering that the recommended resolution is 50 m, it is one of our most challenging benchmarks to run. The full specification is on our website at http://scecdata.usc.edu/cvws/tpv29_30docs.html.

It was my job to design and test these benchmarks. I wrote the code to create the rough fault surface, and generated several of them for testing. I performed about 70 dynamic rupture simulations to choose one of the rough surfaces, and select parameters and benchmark features. Tests were done using my own finite-element code, named FaultMod. Nucleation is done using a method I devised, which combines smoothed forced rupture with time-weakening. It took considerable effort to find a rough surface and parameters that were suitable for both elastic and viscoplastic materials; allowing both cases to propagate well and show clear effects of roughness on the rupture contours, without requiring excessively high numerical resolution.

Material properties are the only difference between TPV29 and TPV30, so that we can directly observe how viscoplasticity affects the results. Figure 3 shows rupture contours for both benchmarks, superimposed on the fault topography. In general, the viscoplastic rupture is more responsive to the roughness than the elastic rupture.

Figures 4 and 5 show rupture contours from the participating modelers, superimposed to illustrate the level of agreement. In the elastic case the agreement is very good, while in the viscoplastic case one or two codes deviate a bit from the rest. The figures also show metrics that quantify the level of agreement between the codes, using the quantitative metrics that I designed (see Barall and Harris, SRL 2015).

Figure 6 shows the on-fault stations, superimposed on the rupture contours. Modelers submit slip history at each on-fault station, so we attempted to select locations that are "interesting". Figure 6 shows an example of two stations which are only 300 meters apart, and yet, in the viscoplastic case, it takes 2 seconds for the rupture to propagate from one station to the other due to the roughness. Figure 7 compares the results from five codes during the 2-second pause.

This year, for the first time, we are placing off-fault stations at a substantial distance from the fault. Modelers submit synthetic seismograms for these stations. Figure 8 shows the locations of off-fault stations for TPV29-30, with the most distant stations 20 km from the fault. Figure 9 uses my quantitative metrics to demonstrate that the codes agree more poorly at the more-distant stations. Figure 10 shows waveforms (synthetic seismograms) for six participating codes, again showing that agreement worsens at the more-distant stations.

Additional results from these benchmarks are described in my workshop presentations, which are available at http://scecdata.usc.edu/cvws/mar23_2015_presentations.html.

In my role as one of the modelers, I uploaded a set of results which can be viewed in the web-based comparison tool at http://scecdata.usc.edu/cvws/cgi-bin/cvws.cgi.

After all the participating modelers submitted their results, it was my job to compare the results and present my analysis at our group workshop, which was held on March 23, 2015.

Vertical Planar Fault in a 1D Velocity Structure Benchmarks

The SCEC Dynamic Rupture Code Verification Group created two benchmarks to test the performance of the dynamic rupture codes in modeling a fault embedded in a stratified medium:

- TPV31 — Discontinuous 1D velocity model, minimum V_S = 2250 m/s.
- TPV32 — Continuous 1D velocity model, minimum V_S = 1050 m/s.

Both benchmarks use a right-lateral strike-slip fault embedded in an elastic halfspace with a 1D velocity structure. Figure 11 shows the fault geometry Figure 12 shows the velocity structure for TPV31, which includes discontinuities at depths of 2.4 km, 5 km, and 10 km. Figure 13 shows the velocity structure for TPV32, which is continuous but also reaches very low velocities near the earth's surface. Due to the low velocities, codes must use high resolution when running TPV32. In fact, TPV32 is our first benchmark where many codes had to use 25 m resolution to obtain acceptable results, making this benchmark very computationally challenging. The full specification is on our website at http://scecdata.usc.edu/cvws/tpv31_32docs.html.

It was my job to design and test these benchmarks. I create the velocity structures, which are loosely based on the Loma Prieta velocity structure. I performed about 40 dynamic rupture simulations to select parameters and benchmark features. Tests were done using my own finite-element code, named FaultMod. Nucleation is done by setting the initial shear stress to be slightly higher than the yield stress within a nucleation zone.

Due to the very low velocities in TPV32, when I ran it I had to use 25 m elements adjacent to the fault, and 50 m elements in the uppermost 1 km of the entire model domain. This made my final tests of TPV32 the largest dynamic rupture simulations I have ever run.

Figures 14 and 15 show rupture contours from the participating modelers, superimposed to illustrate the level of agreement. For both benchmarks, the agreement is excellent, with the contours from different modelers lying right on top of each other. The figures also show metrics that quantify the level of agreement between the codes, using the quantitative metrics that I designed (see Barall and Harris, SRL 2015).

I used my quantitative metrics to perform a study of process zone width, expecting that the velocity models would have a substantial influence. The results shown in figures 16 and 17 confirm this. There is good agreement among the participating codes on the width of the process zone. The width shrinks near the earth's surface where velocities are low. For TPV31, the minimum width is about 200 m, indicating that 50 m numerical resolution should be sufficient. But for TPV32, which has much lower velocity, the minimum width is about 70 m, explaining why 25 m numerical resolution is barely enough.

Figure 18 shows slip rate reported by 9 participating codes at the epicenter, and at a station 200 m below the epicenter, for TPV32. Due to the very low velocities and small process zone widths, it is challenging to get good results at these stations. The codes exhibit some amount of disagreement and oscillation. Also, the waveforms at the two stations are quite different despite the fact that the stations are only 200 m apart.

Figure 19 shows the location of off-fault stations, where modelers submit synthetic seismograms. This year for the first time we are placing stations far from the fault, and so stations are located 3 km, 9 km, and 15 km away from the fault. Also new is that the stations are organized into "boreholes" with stations at depths of 0.0 km, 0.5 km, and 2.4 km, so that we can observe how the motions changes in the low-velocity zones near the earth's surface.

Figure 20 shows the waveforms (synthetic seismograms) at different depths. For TPV31, which has constant velocity in the uppermost 0.5 km, there is little difference between the stations at 0.0 and 0.5 km depth, although both of these have considerably higher peak velocity than at 2.4 km depth. In contrast, for TPV32, which has sharply decreasing velocity in the uppermost 0.5 km, the station at 0.0 km depth has much higher peak velocity than at 0.5 km depth, which demonstrates the "amplification" effect of the low-velocity layer.

Figure 21 shows examples of waveforms (synthetic seismograms) for TPV32, produced by 8 participating codes, at distances of 3 km and 15 km from the fault. Agreement among the codes deteriorates noticeably at the more distant station.

Additional results from these benchmarks are described in my workshop presentations, which are available at http://scecdata.usc.edu/cvws/mar23_2015_presentations.html.

In my role as one of the modelers, I uploaded a set of results which can be viewed in the web-based comparison tool at http://scecdata.usc.edu/cvws/cgi-bin/cvws.cgi.

After all the participating modelers submitted their results, it was my job to compare the results and present my analysis at our group workshop, which was held on March 23, 2015.

Performance of Triangular Fault Elements in Earthquake Simulators

This is a new project that I performed with co-PI Terry Tullis (project number 14090).

An earthquake simulator is a computer program that generates a synthetic earthquake catalog spanning thousands of years, or longer. Most of the computational effort in an earthquake simulator goes into computing how slip on one part of a fault affects stresses on other parts of the fault, and on other faults. The computation is done by discretizing the fault system into a large number of fault elements, and using Greens functions to determine how a pattern of slip on some fault elements affects the stresses on all the fault elements.

Traditionally, earthquake simulators have used rectangular fault elements, chosen so that the Okada Greens functions can be used. Recently, due to the development of new Greens functions for triangular dislocations, it has become practical to use triangular fault elements (Meade 2007; Gimbutas *et al.* 2012). The purpose of this project is to assess the accuracy of stress calculations performed with triangular fault elements, as compared to the accuracy of the same calculations done with rectangular fault elements.

Because triangles can represent curved fault geometry more accurately than rectangles, one intuitively expects that stress calculations performed with triangles should be more accurate than stress calculations done with rectangles. The original motivation for this project was to demonstrate that this intuitive expectation is true, and quantify the difference in accuracy.

However, our results are contrary to the intuitive expectation. In our tests, triangles are not superior to rectangles. One or the other may be superior in a particular case, but, overall, rectangles perform as well as or better than triangles. Another unexpected result is that one triangulation of a fault surface may perform significantly better than another triangulation with a different pattern of triangles.

To test the performance of triangular fault elements, I created two fault surfaces, one with negative curvature and one with positive curvature. The negatively-curved surface is motivated by the shape of the San Andreas Fault in southern California (Fuis *et al*. 2012). It is a strike-slip fault in the shape of a helicoid, as shown in figure 22. The positive-curved surface is motivated by the shape of the Cascadia subduction fault (McCrory *et al.* 2012). It is a thrust fault in the shape of an ellipsoid, and shown in figure 23.

I wrote a computer program to create triangulations and rectangulations of each fault surface. Using that program, I discretized each fault surface in three different ways: one pattern of rectangular elements, and two different patterns of triangular elements. This was repeated in each of six different element sizes. Figures 24 and 25 show the discretizations. In this way, I can compare not only rectangles to triangles, but also the two different triangulations.

Our strategy is to impose uniform slip on part of the fault surface, called the *source region*, and then compute the induced shear and normal stresses elsewhere on the fault surface, called the *target region*. Figures 26 and 27 show the source and target regions I used. In some cases, the target region borders on the source region; while in other cases the target region is 4.8 km away from the source region. The former case is relevant for the simulation of rupture propagation during an earthquake, because the earthquake rupture proceeds from the currently-slipping elements to their neighboring elements. The latter case is relevant for examining the accuracy of stress transfer over larger distances. We chose a separation of 4.8 km because it is roughly the maximum size of a fault-to-fault jump (Wesnousky 2006).

To assess the accuracy of the calculation, we discretize the fault surfaces using either rectangles or triangles, each in six different sizes: 300, 600, 1200, 2400, 4800, and 9600 meters. The 300 m results are taken as the reference solution, and the accuracy of the other five sets of results is assessed by how well they approximate the reference value.

I defined three different accuracy tests:

1. A *source test* which evaluates the error originating at the source elements. This test is performed using a source region and target region that are separated by 4800 meters. We vary the discretization of the source region, using either triangles or rectangles with sizes ranging from 600 to 9600 m. The discretization of the target region is held fixed.

2. A *target test* which evaluates the error originating at the target elements. This test is also performed using a source region and target region that are separated by 4800 meters. The discretization of the source region is held fixed. We vary the discretization of the target region, using either triangles or rectangles with sizes ranging from 600 to 9600 m.

3. A *propagation test* which evaluates errors occurring during rupture propagation. This test is performed using a target region that borders on the source region, with no separation. We vary the discretization of both the source region and the target region, using either triangles or rectangles with sizes ranging from 600 to 9600 m. The source and target regions are discretized in the same way (e.g., if one is 1200 m triangles then so is the other), as would be the case in a simulator.

I ran all the tests, and graphed the results. A few of the resulting graphs are shown in figures 28 to 31. The graphs show how the accuracy of the stress calculation varies as the size of the fault elements is varied. There are separate graphs for shear stress and normal stress.

Each graph is a log-log plot, in which the approximation error, in percent, is plotted against the number of fault elements. The number of fault elements is the right thing to use on the horizontal axis, because the number of fault elements is what determines the computational effort that a simulator must exert to calculate the stress transfer. There are three curves on each plot: a blue curve that shows results for rectangular fault elements, and red and green curves that show results for triangular fault elements. The red and green curves correspond to two different methods for constructing the triangulation.

Examining all the graphs, I formed the following conclusions:

*   On the positively-curved fault, rectangles performed as well as or better than triangles in every test. In comparing the two triangulations, denoted triangle-1 and triangle-3, we found that the two triangulations performed equally well.

*   On the negatively-curved fault, results were mixed. In comparing the two triangulations, denoted triangle-2 and triangle-4, we found that triangle-4 performed better than triangle-2 in every test. In the source test, rectangles performed as well as triangle-4 when calculating shear stress. When calculating normal stress, rectangles were less accurate than triangle-4, but still more accurate than triangle-2. In the target test, rectangles were far more accurate than triangles, for both shear and normal stress. In the propagation test, when calculating shear stress, rectangles achieved the same accuracy as triangle-4 in one test, but were less accurate than triangle-2 in the other test. When calculating normal stress, rectangles were far more accurate than triangles.

The poor performance of triangle-2 is noteworthy, because it is the simplest method of constructing a triangulation and so it is frequently used.

Our overall conclusion is that in an earthquake simulator, rectangles should perform as well as triangles.

Finally, I wrote the project report. The report, which is available in the SCEC project reports database, provides many more details and results.

References

Barall, M. and R.A. Harris (2015), Metrics for Comparing Dynamic Earthquake Rupture Simulations, *Seismological Research Letters*, v. 86, n. 1, p. 223-235, doi:10.1785/0220140122.

Fuis, G.S., D.S. Scheirer, V.E. Langenheim, and M.D. Kohler (2012), A New Perspective on the Geometry of the San Andreas Fault in Southern California and Its Relationship to Lithospheric Structure, *Bulletin of the Seismological Society of America*, v. 102, n. 1, p. 236-251, doi:10.1785/0120110041.

Gimbutas, Z., L. Greengard, M. Barall, and T.E. Tullis (2012), On the Calculation of Displacement, Stress, and Strain Induced by Triangular Dislocations, *Bulletin of the Seismological Society of America*, v. 102, n. 6, p. 2776-2780, doi:10.1785/0120120127.

McCrory, P.A., J.L. Blair, F. Waldhauser, and D.H. Oppenheimer (2012), Juan de Fuca slab geometry and its relation to Wadati-Benioff zone seismicity, *Journal of Geophysical Research*, v. 117, B09306, doi:10.1029/2012JB009407.

Meade, B.J. (2007), Algorithms for the calculation of exact displacements, strains, and stresses for triangular dislocation elements in a uniform elastic half space, *Computers & Geosciences*, v. 33, n. 8, p. 1064-1075, doi: 10.1016/j.cageo.2006.12.003.

Wesnousky, S.G. (2006), Predicting the endpoints of earthquake ruptures, *Nature*, v. 444, n. 7117, p. 358-360, doi: 10.1038/nature05275.

Figures
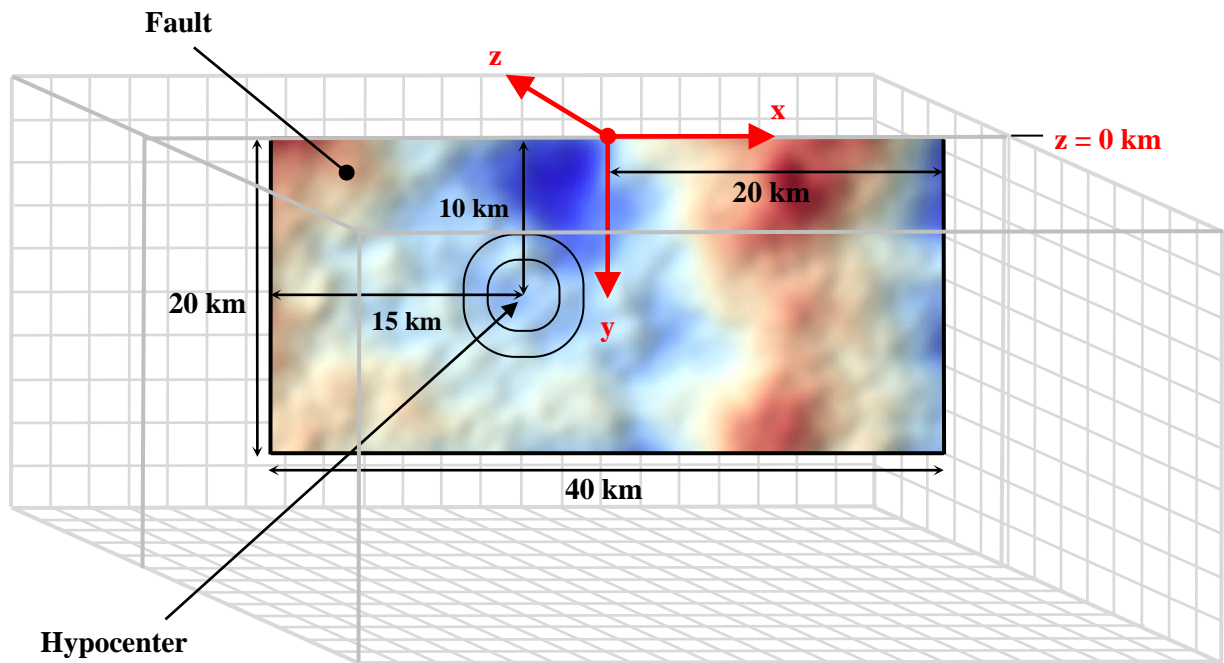
TPV29 uses linear elastic material:

    Density $\rho = 2670 \text{ kg/m}^3$

    Shear-wave velocity $V_S = 3464$ m/s

    Pressure-wave velocity $V_P = 6000$ m/s

TPV30 uses viscoplastic material:

    Density $\rho = 2670 \text{ kg/m}^3$

    Shear-wave velocity $V_S = 3464$ m/s

    Pressure-wave velocity $V_P = 6000$ m/s

    Cohesion $c = 1.18$ MPa

    Bulk friction $\nu = 0.1680$

    Relaxation time $T_\nu = 0.05$ s

Figure 1. Geometry and material properties for the rough fault with viscoplasticity benchmarks. TPV29 is set in an elastic halfspace. TPV30 is set in a Drucker-Prager viscoplastic halfspace. Both benchmarks use a vertical, right-lateral, strike-slip fault with stochastic roughness.
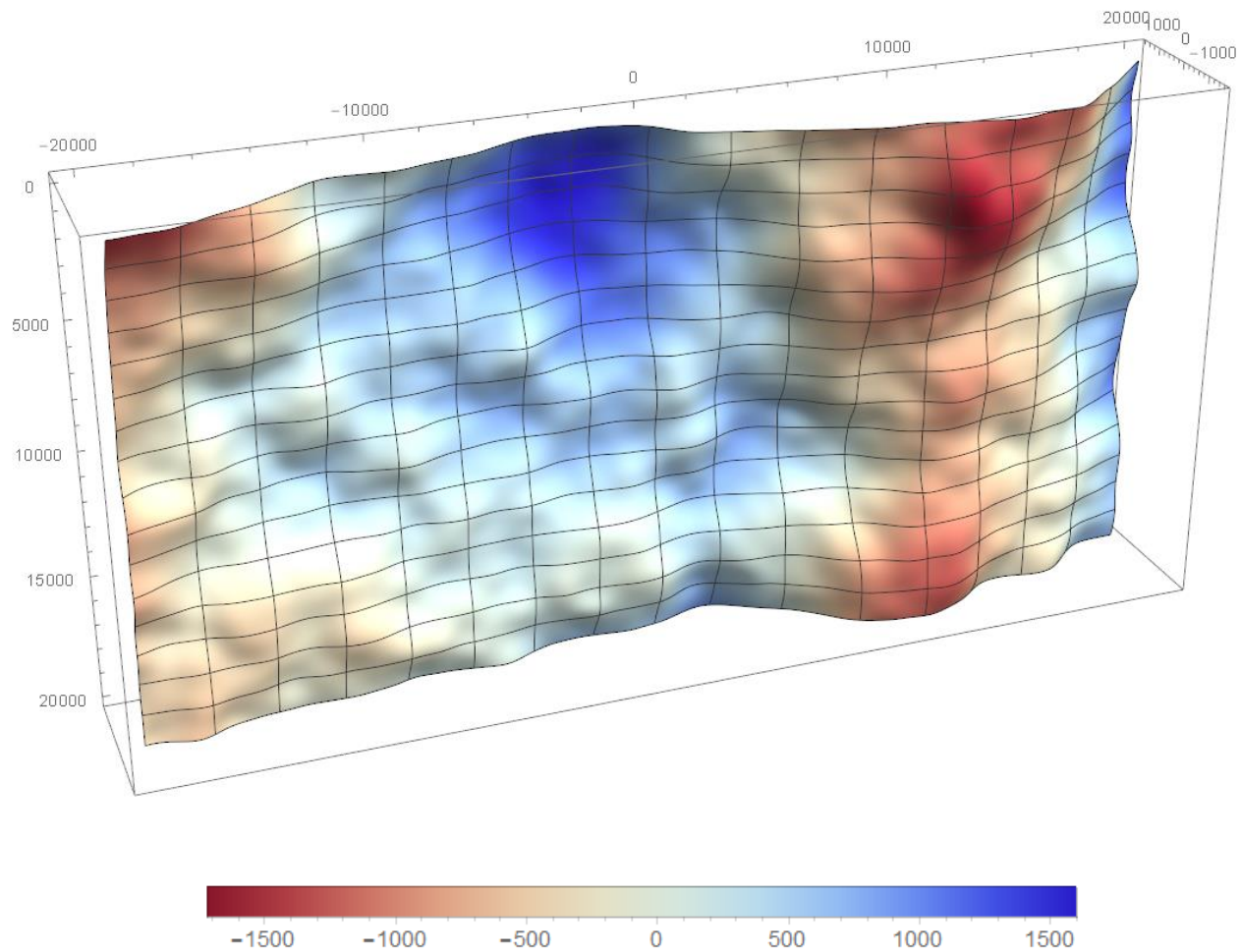
Figure 2. Rough-fault topography used for benchmarks TPV29 and TPV30. Note that relief is exaggerated in the image. The fault roughness is self-similar, with Hurst exponent H=1. It is constructed using a Fourier transform technique:

1. Begin with a square region, 400 km on a side, gridded at 25 m resolution.
2. Working in the frequency domain, each Fourier coefficient with wavelength between 1,000 m and 40,000 m is assigned a value proportional to $k^{-(H+1)}$ and a random complex phase, where k is the wavenumber.
3. Perform a Fourier transform to get the fault roughness in the spatial domain.
4. Cut out an arbitrarily-chosen region the size of the fault surface.
5. Apply a moving-average filter, using a 2D filter kernel with a cosine shape and a half-width of 1,000 m.

You can think of the fault roughness as being a linear superposition of sine waves, each with random phase. The sine waves have wavelengths ranging from 1,000 m to 40,000 m, and amplitude proportional to the square of the wavelength.

10
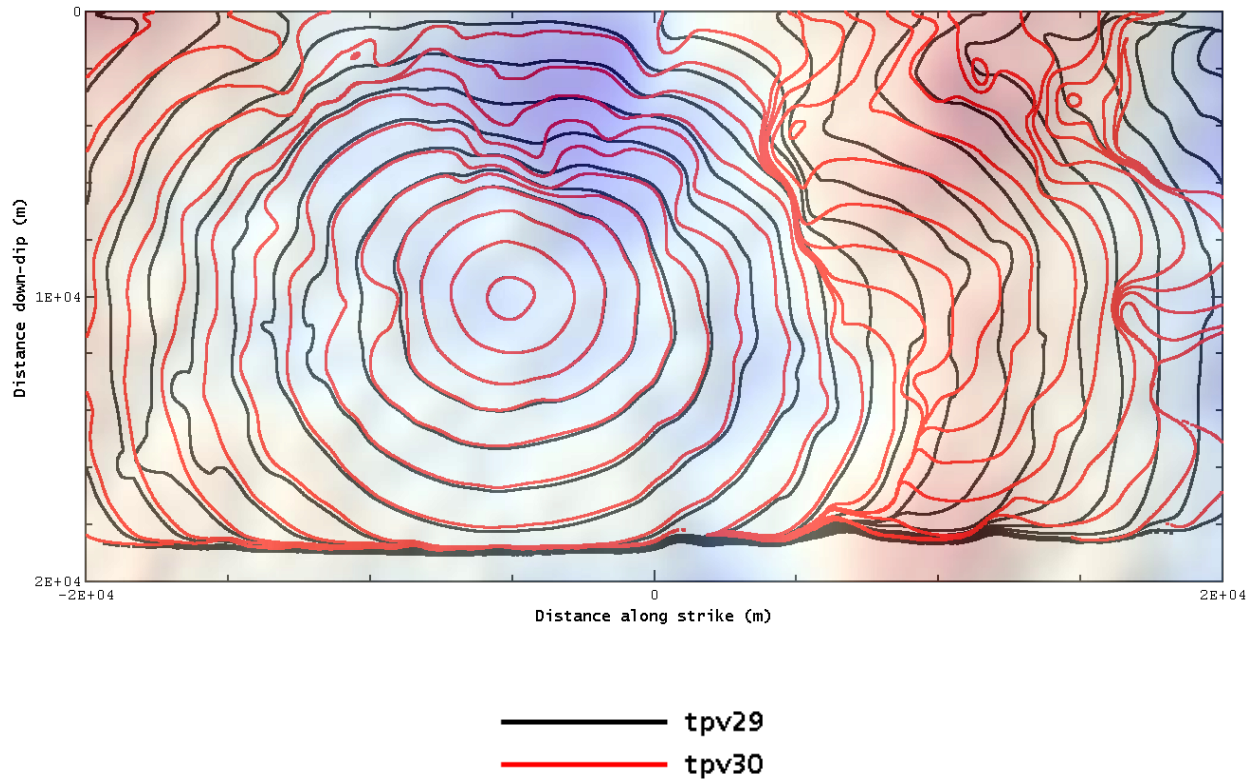
Figure 3. Rupture contours for TPV29 (elastic case) and TPV30 (viscoplastic case), superimposed on the fault topography. The image shows the rupture fronts, at intervals of 0.5 seconds. There is clearly some correlation between rupture contours and fault topography, but it is difficult to look at the fault topography and predict what the rupture will do. The viscoplastic case (red contours) is more responsive to the fault roughness.

bai (Kangchen Bai - Spectral Element - SPECFEM3D 50m)
barall.2 (Michael Barall - FaultMod - 25 m)
dliu.2 (Dunyu Liu - Finite Element - EQdyna-50m)
duru.2 (Kenneth Duru - WaveQLab3D - 50 m)
gabriel (Thomas Ulrich - Discontinous Galerkin- Seissol 100m)
kozdon (Jeremy Kozdon - beard :: p = 4 :: 100 m (avg) :: Hermite interp)
ma.2 (Shuo Ma - Finite Element - MAFE - 50m)
shi.2 (Zheqiang Shi - SORD-S - 25 m)
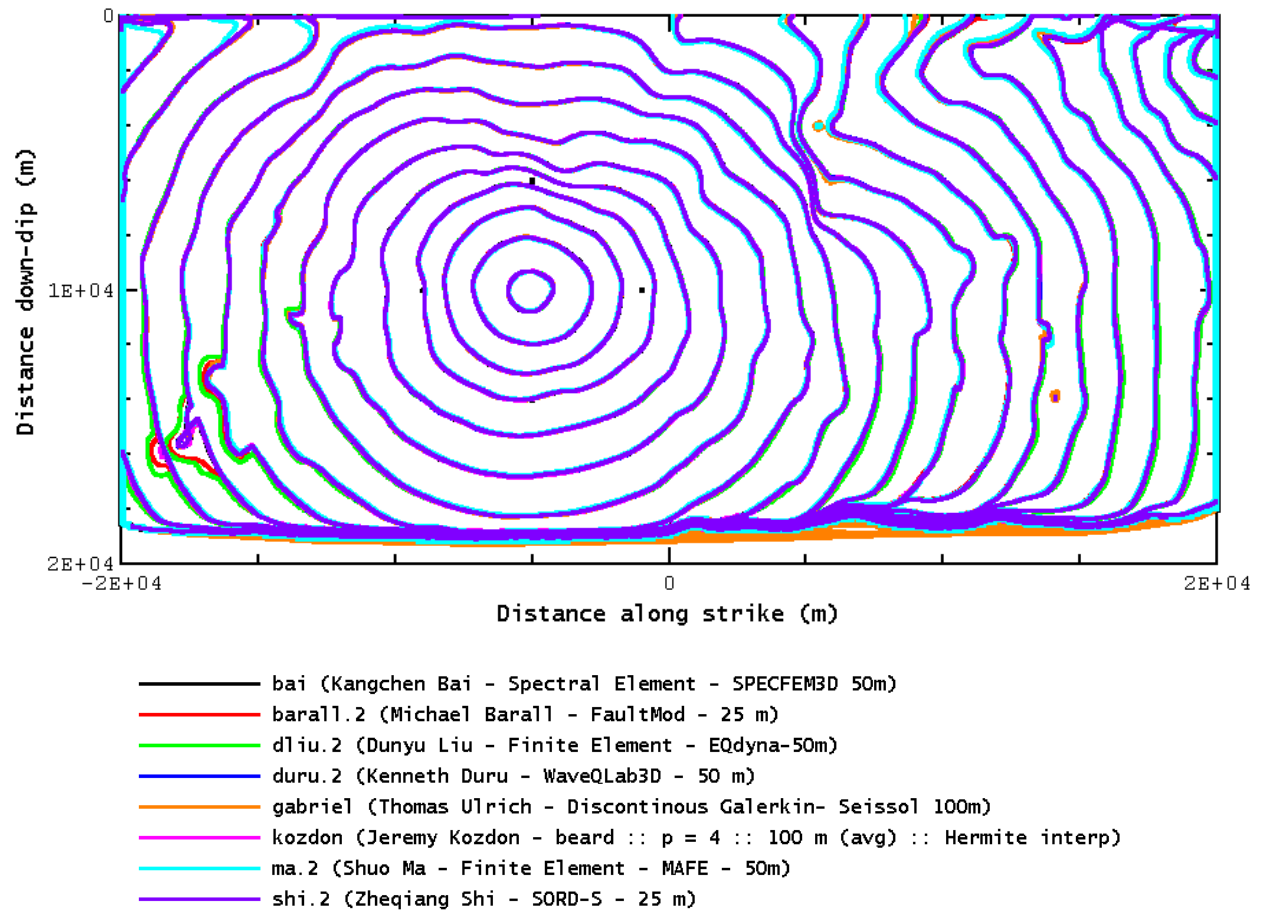
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
(1) bai		19.6	30.6	8.0	15.6	7.8	30.0	10.7
(2) barall.2	19.6		21.4	22.1	25.7	21.4	37.0	23.3
(3) dliu.2	30.6	21.4		32.1	36.7	31.2	48.8	33.5
(4) duru.2	8.0	22.1	32.1		15.4	5.5	30.4	7.6
(5) gabriel	15.6	25.7	36.7	15.4		15.3	22.3	15.7
(6) kozdon	7.8	21.4	31.2	5.5	15.3		30.0	9.3
(7) ma.2	30.0	37.0	48.8	30.4	22.3	30.0		28.6
(8) shi.2	10.7	23.3	33.5	7.6	15.7	9.3	28.6	

Figure 4. TPV29 (elastic case) rupture contours from 8 participating modelers, superimposed to demonstrate very good agreement between the codes. The table shows the RMS difference in rupture time, in milliseconds, between any pair of codes.

12
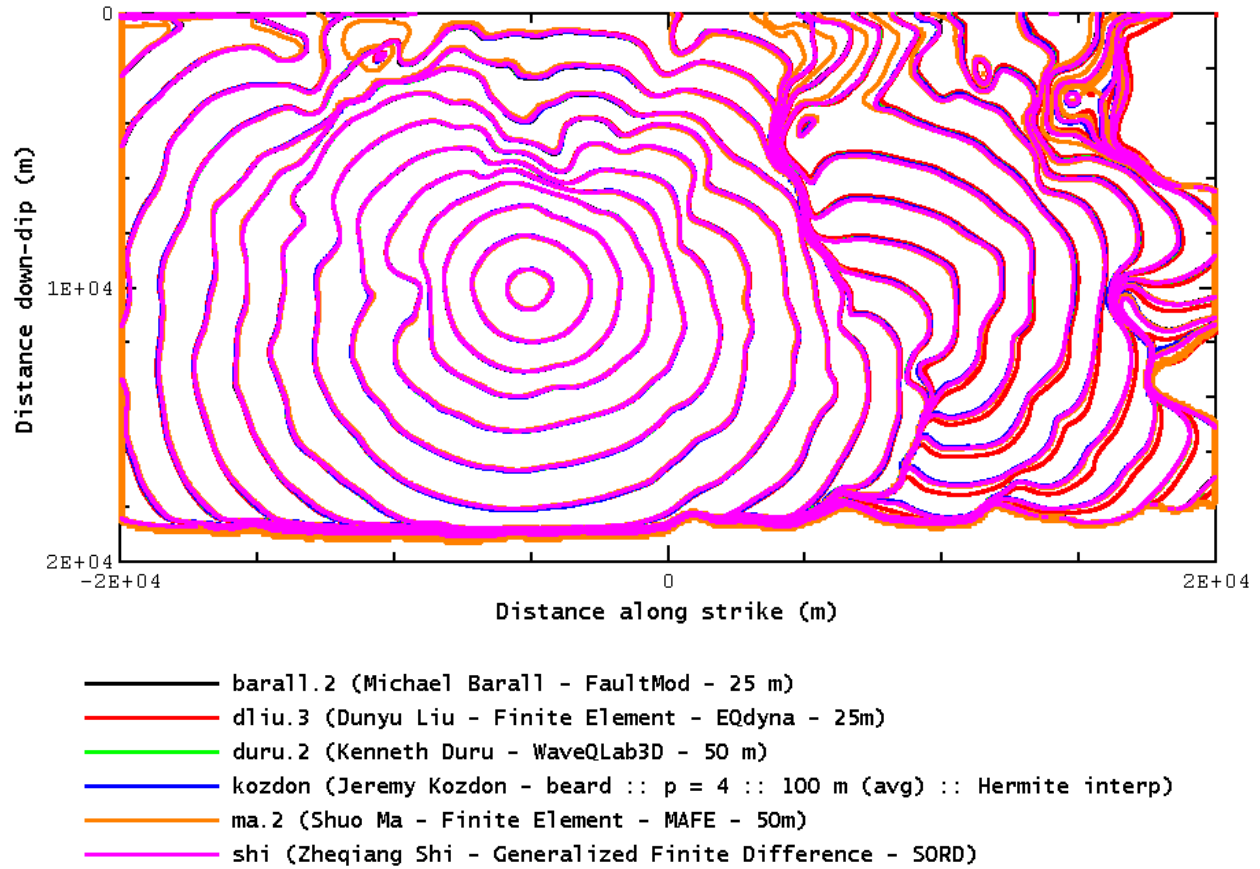
	(1)	(2)	(3)	(4)	(5)	(6)
(1) barall.2		46.8	9.6	9.4	62.4	9.6
(2) dliu.3	46.8		40.2	45.0	82.3	45.4
(3) duru.2	9.6	40.2		7.2	62.9	11.8
(4) kozdon	9.4	45.0	7.2		61.5	13.3
(5) ma.2	62.4	82.3	62.9	61.5		59.0
(6) shi	9.6	45.4	11.8	13.3	59.0	

Figure 5. TPV30 (viscoplastic case) rupture contours from 6 participating modelers, superimposed to illustrate agreement between the codes. Rupture contours are plotted at intervals of 0.5 seconds. Four of the codes agree very well; the other two deviate from the rest at the top and lower-right parts of the fault. The table shows the RMS difference in rupture time, in milliseconds, between any pair of codes. Generally, we regard values less than 50 ms to be reasonably good; two of the codes have disagreements that are consistently close to or exceeding 50 ms.
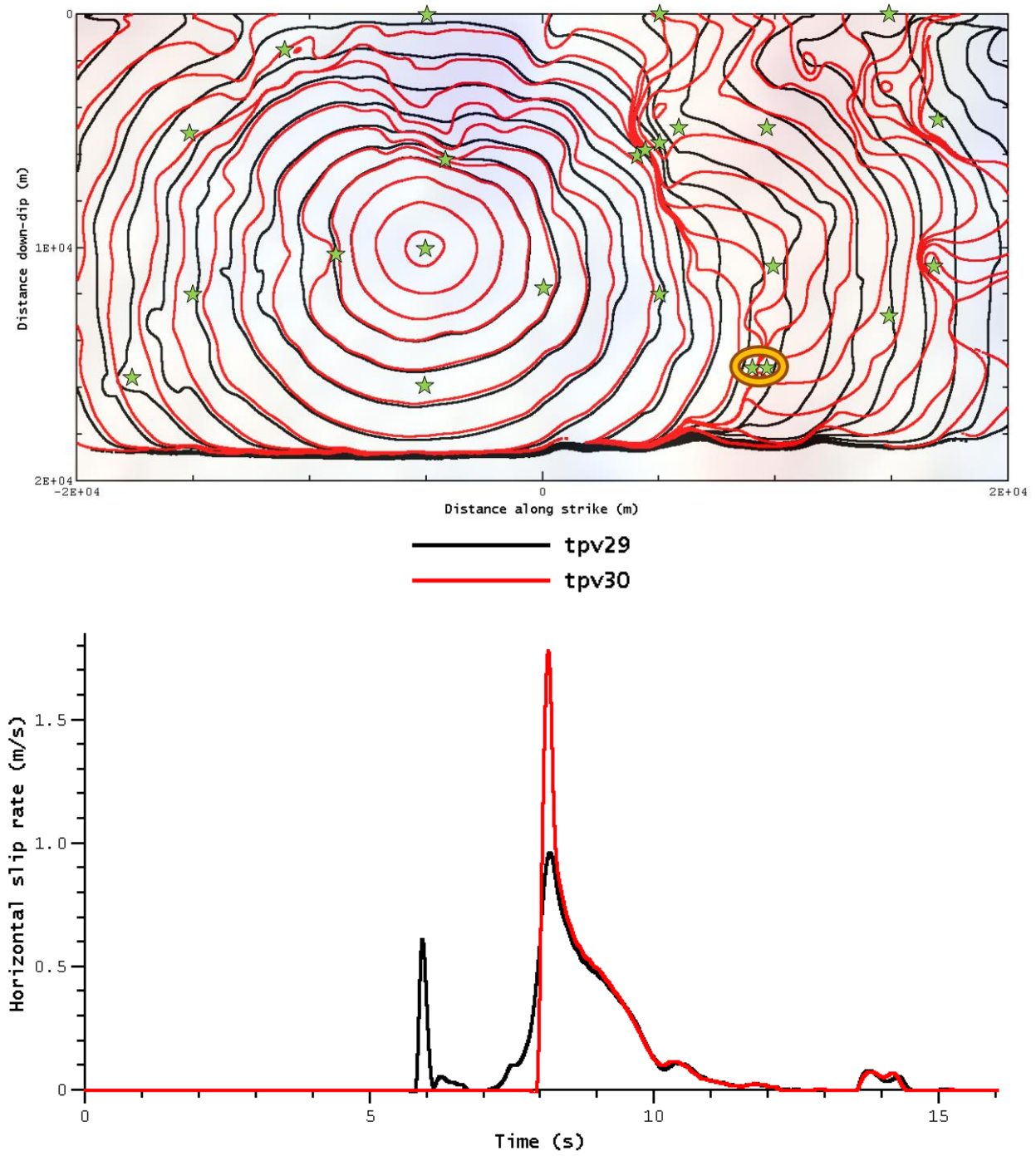
Figre 6. Top: Locations of on-fault stations, shown as green stars, superimposed on the rupture contours. We attempted to choose "interesting" locations. Bottom: Horizontal slip rate, for the two circled stations, in TPV30 (viscoelastic case). The stations are only 300 m apart, yet it takes 2 seconds for the rupture to propagate from one to the other. Black curve is for the first (left) station, and shows an initial pulse, followed by a 2-second pause. Red curve is for the second (right) station, showing no movement until after the pause.

14
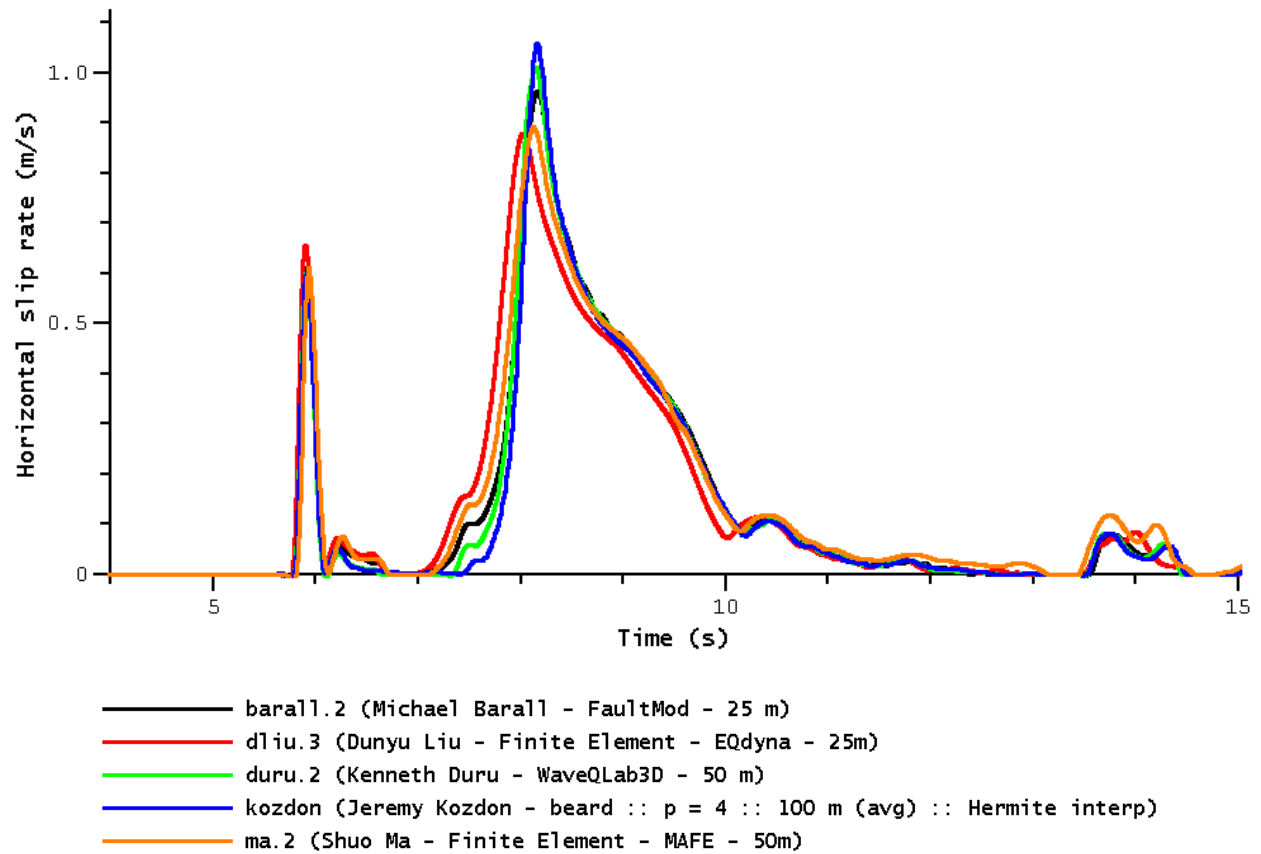
Figure 7. Horizontal slip rate for 5 participating codes, at the first (left) station identified in figure 6. Codes agree very well during the initial pulse. As slip resumes at the end of the pause, there is some disagreement between the codes.
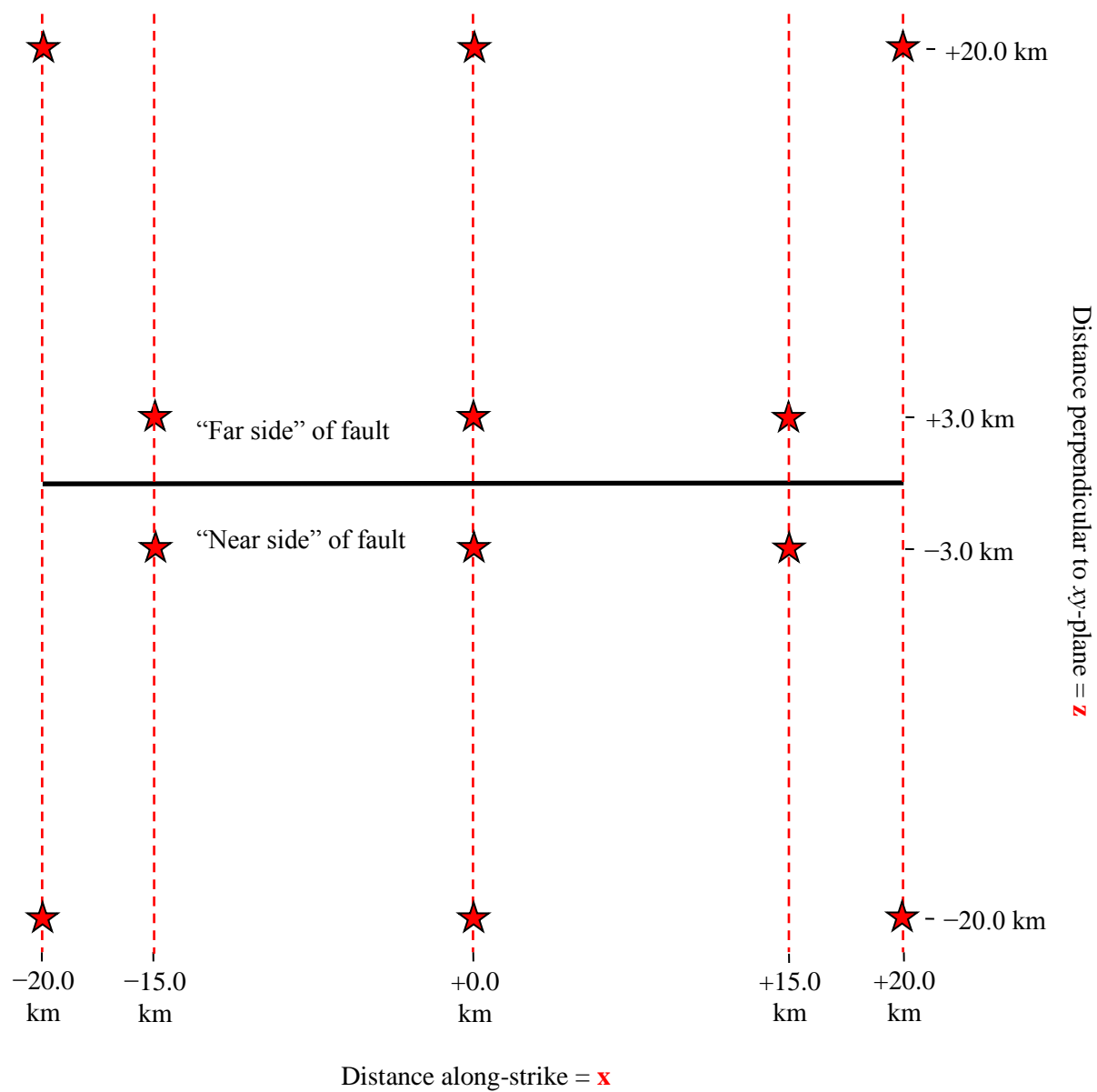
Figure 8. Locations of off-fault stations for benchmarks TPV29-30. The figure shows station locations on the earth's surface, in map view. This year, for the first time, we are placing off-fault stations far from the fault. Here, the most distant stations are 20 km from the fault.

	3d-vel	5 codes
body-030st-150dp000	5.3	3.9
body-030st000dp000	4.7	3.2
body-030st150dp000	10.2	6.3
body030st-150dp000	6.8	5.0
body030st000dp000	6.1	4.2
body030st150dp000	11.5	8.1

	3d-vel	5 codes
body-200st-200dp000	14.4	9.7
body-200st000dp000	13.9	9.0
body-200st200dp000	23.0	10.5
body200st-200dp000	15.4	11.2
body200st000dp000	13.4	7.8
body200st200dp000	27.3	13.5

	3d-vel
body-030st-150dp000	6.4
body-030st000dp000	6.5
body-030st150dp000	10.9
body030st-150dp000	8.4
body030st000dp000	7.4
body030st150dp000	11.8

	3d-vel
body-200st-200dp000	18.6
body-200st000dp000	12.9
body-200st200dp000	18.6
body200st-200dp000	18.8
body200st000dp000	13.1
body200st200dp000	20.1

Figure 9. Metric values for off-fault stations in TPV29-30. Lower values indicate better agreement among the codes. The top two tables show metrics for TPV29 (elastic case); the "3d-vel" column shows the average metric for 3D velocity among the participating codes, and the "5 codes" column shows the same for the five best-agreeing codes. The bottom two tables show metrics for TPV30 (viscoplastic case). In each case, the table on the left is for stations 3 km from the fault, and the table on the right is for stations 20 km from the fault. In each case, agreement is significantly worse for the stations 20 km from the fault.

17
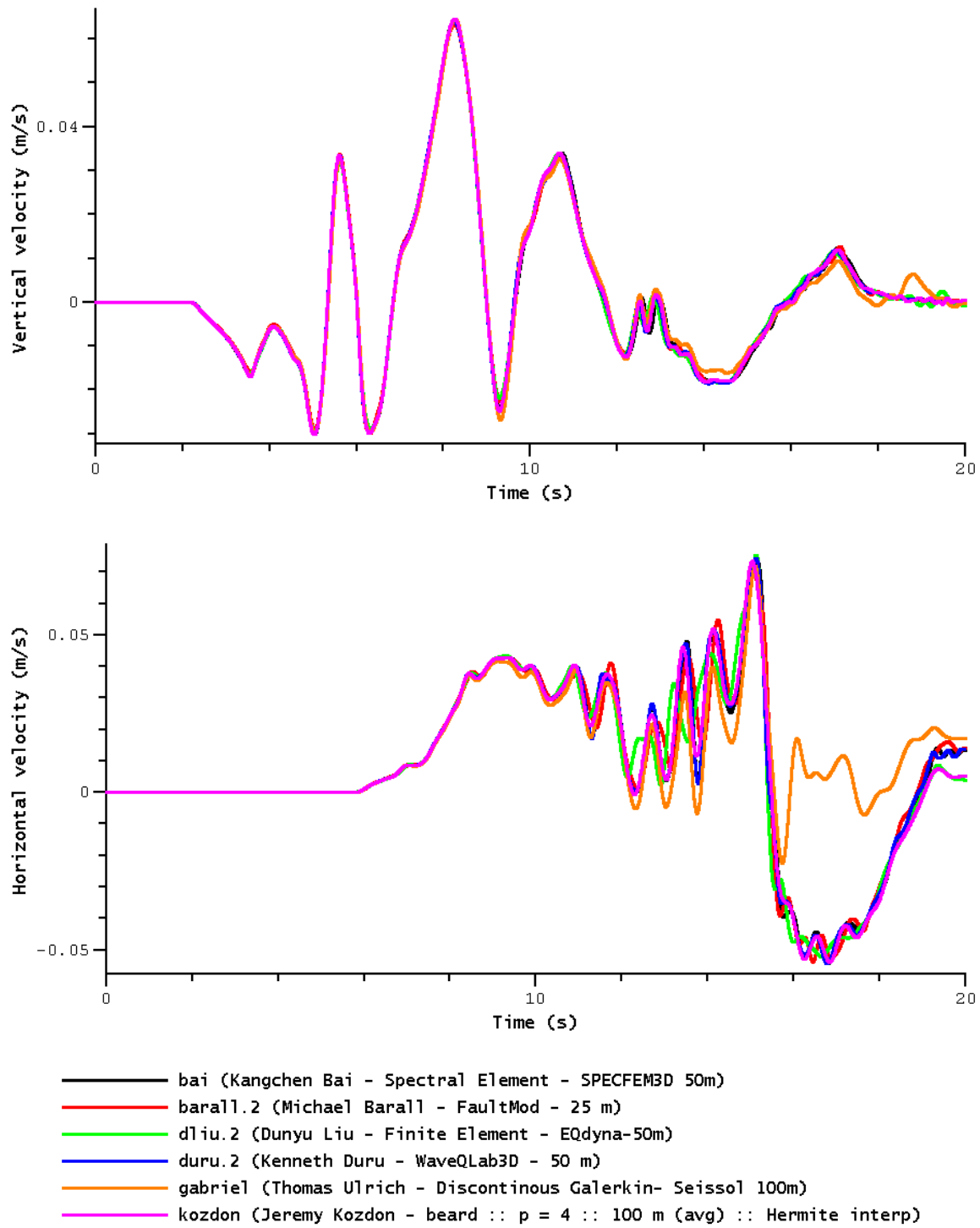
Figure 10. Waveforms (synthetic seismograms) for TPV29 for 6 participating codes. The top figure is for a station 3 km from the fault, and the bottom figure is for a station 20 km from the fault. It is evident that agreement among the codes worsens at the more-distant station.

18
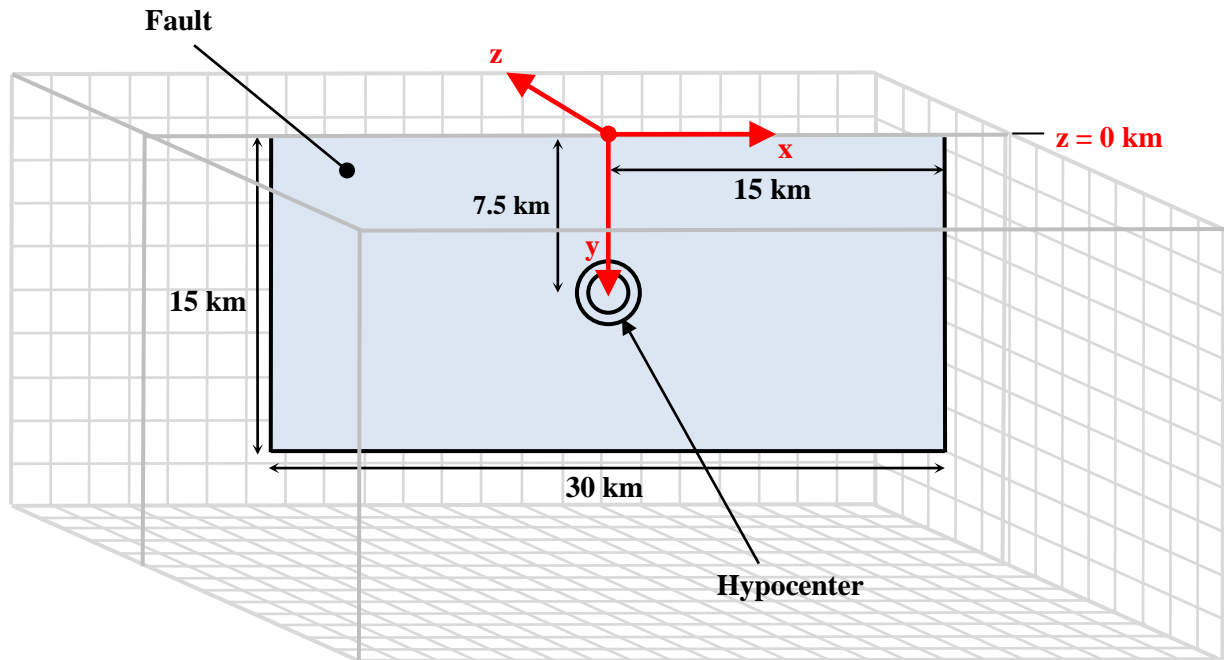
Right-lateral strike-slip fault in an elastic half-space with a 1D velocity structure.

TPV31 — Discontinuous 1D velocity model, minimum V_S = 2250 m/s.

TPV32 — Continuous 1D velocity model, minimum V_S = 1050 m/s.

Figure 11. Geometry and material properties for the 1D velocity structure benchmarks. TPV31 is set in an elastic halfspace with a discontinuous velocity structure. TPV32 is set in an elastic halfspace with a continuous velocity structure. Both benchmarks use a vertical, planar, right-lateral, strike-slip fault.
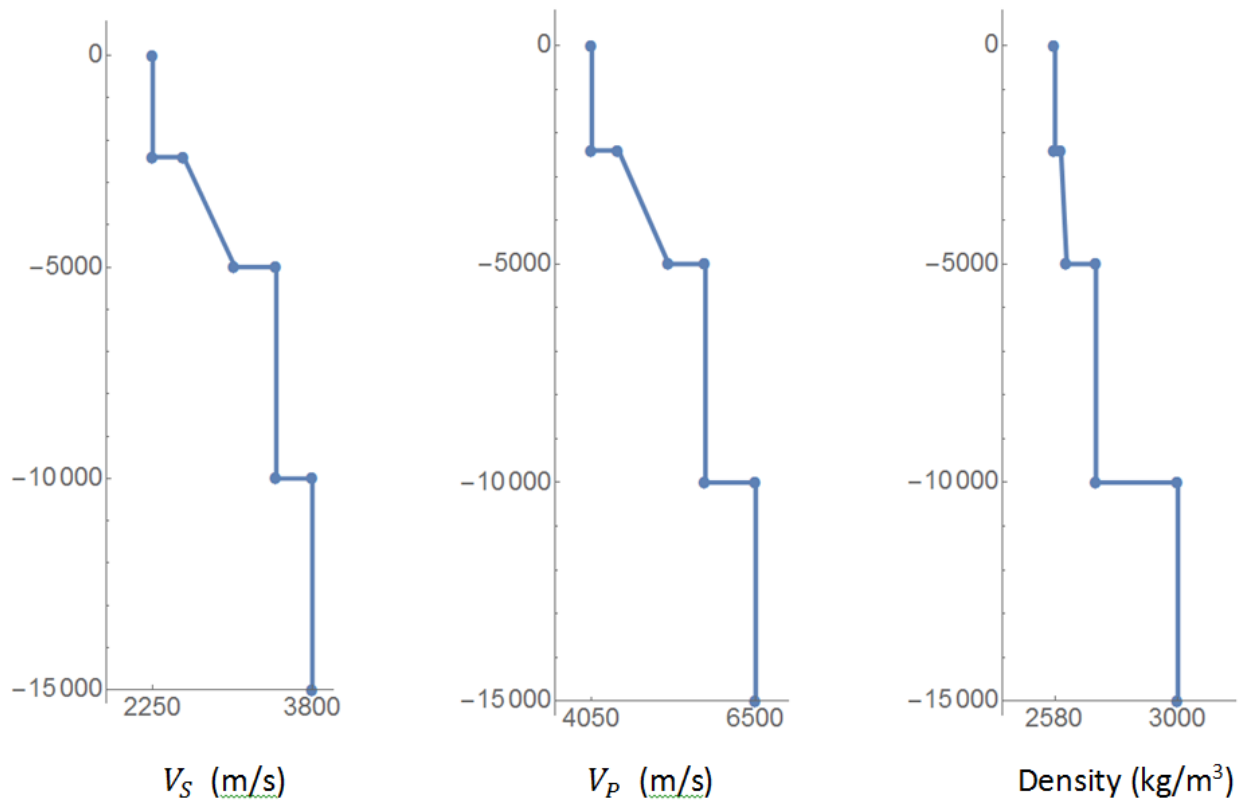
Figure 12. Velocity structure for TPV31. Notice the discontinuities at depths of 2.4 km, 5 km, and 10 km.
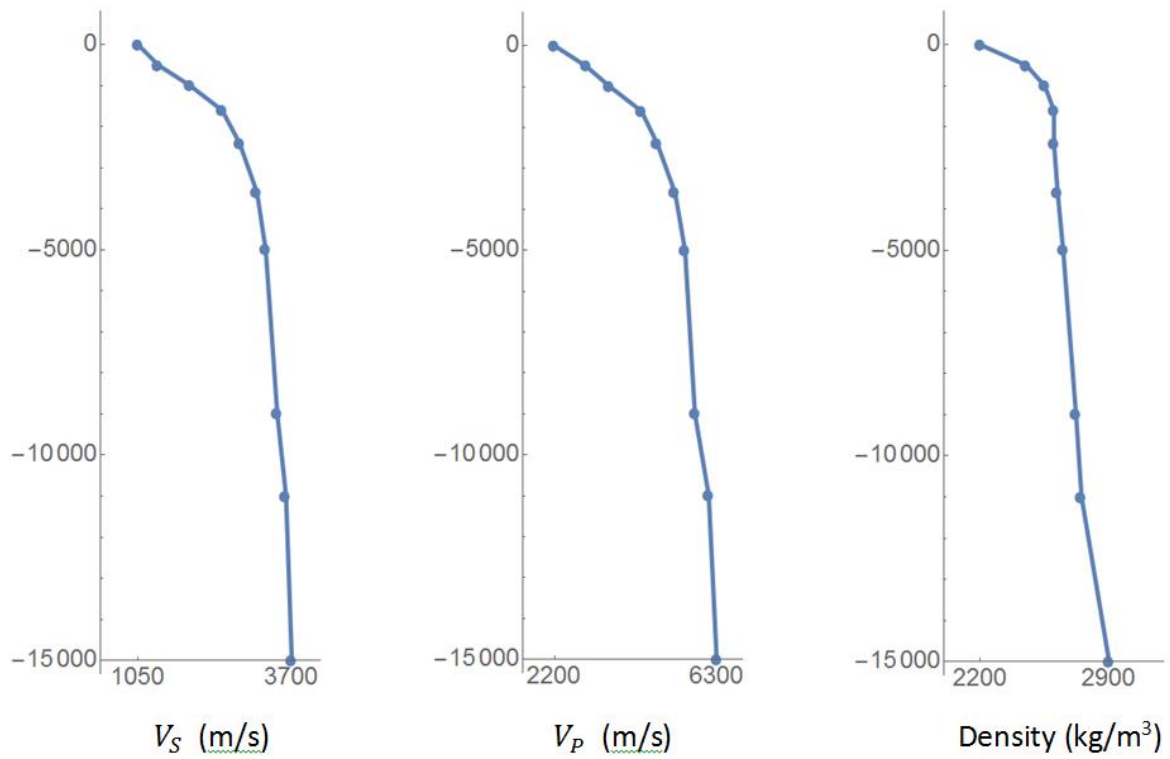
Figure 13. Velocity structure for TPV32. Notice that the velocity structure is continuous, and that it reaches very low velocity at the earth's surface.

Figure 14. TPV31 (discontinuous case) rupture contours from 9 participating modelers, superimposed to demonstrate excellent agreement between the codes. The table shows the RMS difference in rupture time, in milliseconds, between any pair of codes. Note the bends in the contours at depths where the velocity structure is discontinuous: 2.4 km, 5 km, and 10 km.

Figure 15. TPV32 (continuous, low-V_S case) rupture contours from 9 participating modelers, superimposed to demonstrate excellent agreement between the codes. The table shows the RMS difference in rupture time, in milliseconds, between any pair of codes. Note that the rupture contours are smooth, and refract (become near-horizontal) near the earth's surface.

23

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
faultst060dp000	210	211	212	208	209	205	207	199	207
faultst060dp002	280	277	274	276	273	275	273	280	276
faultst060dp005	345	344	343	345	346	349	332	349	348
faultst060dp010	326	318	320	325	324	334	298	331	317
faultst060dp024	292	287	284	290	287	302	266	297	281
faultst060dp030	322	322	317	324	319	333	309	326	316
faultst060dp050	418	418	415	418	415	427	395	429	414
faultst060dp075	479	480	477	479	477	485	473	483	477
faultst060dp100	452	451	453	451	451	455	434	461	450
faultst060dp120	414	412	411	411	411	417	402	414	410

(1) bai	Kangchen Bai - Spectral Element - SPECFEM3D	(6) luo.2	Bin Luo - Finite Element - EQdyna - 25m
(2) barall.2	Michael Barall - FaultMod - 25 m	(7) ma.2	Shuo Ma - Finite Element - MAFE - 50m
(3) bydlon.2	Sam Bydlon - WaveQLab3D - 50 m	(8) roten.2	Daniel Roten - Finite Difference - AWM (25m)
(4) kaneko	Yoshihiro Kaneko - SPECFEM3D (older version) - 50 m	(9) shi.2	Zheqiang Shi - SORD-S - 25 m
(5) kozdon	Jeremy Kozdon - beard :: p = 4 :: 100 m (avg)		

Figure 16. Process zone width for TPV31 (discontinuous case, min V_S = 2250 m/s). The table shows the process zone width in meters for each of 9 participating codes, at 10 on-fault stations of varying depth. Station depth is indicated by the last three digits of the station name. Reading from the top, the depths of the stations are 0.0, 0.2, 0.5, 1.0, 2.4, 3.0, 5.0, 7.5, 10.0, and 12.0 km. As shown by the horizontal bands of color, the codes agree well on the width of the process zone. The process zone width gets smaller as one approaches the earth's surface, where velocity is lower. The smallest widths are about 200 m, indicated that this benchmark should be resolvable with a 50 m mesh.

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
faultst060dp000	80	72	81	78	82	79	78	80	67
faultst060dp002	175	166	166	177	171	182	150	180	165
faultst060dp005	169	162	165	168	172	178	149	174	158
faultst060dp010	221	210	210	217	217	226	195	221	205
faultst060dp024	322	323	321	325	322	334	309	327	319
faultst060dp030	354	354	351	355	351	362	342	357	349
faultst060dp050	433	434	432	433	431	440	425	437	431
faultst060dp075	478	479	478	478	475	485	477	481	476
faultst060dp100	461	461	456	462	459	467	456	465	459
faultst060dp120	419	417	416	417	417	424	413	420	415

(1) bai	Kangchen Bai - Spectral Element - SPECFEM3D 50m	(6) luo	Bin Luo - Finite Element - EQdyna - 25m
(2) barall	Michael Barall - FaultMod - 25 m	(7) ma.2	Shuo Ma - Finite Element - MAFE - 50m
(3) bydlon.2	Sam Bydlon - WaveQLab3D - 50 m	(8) roten	Daniel Roten - Finite Difference - AWM (25m)
(4) kaneko	Yoshihiro Kaneko - SPECFEM3D (older version) - 50 m	(9) shi	Zheqiang Shi - SORD-S - 25 m
(5) kozdon	Jeremy Kozdon - beard :: p = 4 :: 100 m (avg)		

Figure 17. Process zone width for TPV32 (continuous case, min V_S = 1050 m/s). The table shows the process zone width in meters for each of 9 participating codes, at 10 on-fault stations of varying depth. Station depth is indicated by the last three digits of the station name. Reading from the top, the depths of the stations are 0.0, 0.2, 0.5, 1.0, 2.4, 3.0, 5.0, 7.5, 10.0, and 12.0 km. As shown by the horizontal bands of color, the codes agree well on the width of the process zone. Near the earth's surface, process zone widths are much smaller than TPV31, due to the lower velocity. The smallest width is about 70 m, indicating that a 25 m mesh is barely enough to resolve this benchmark.
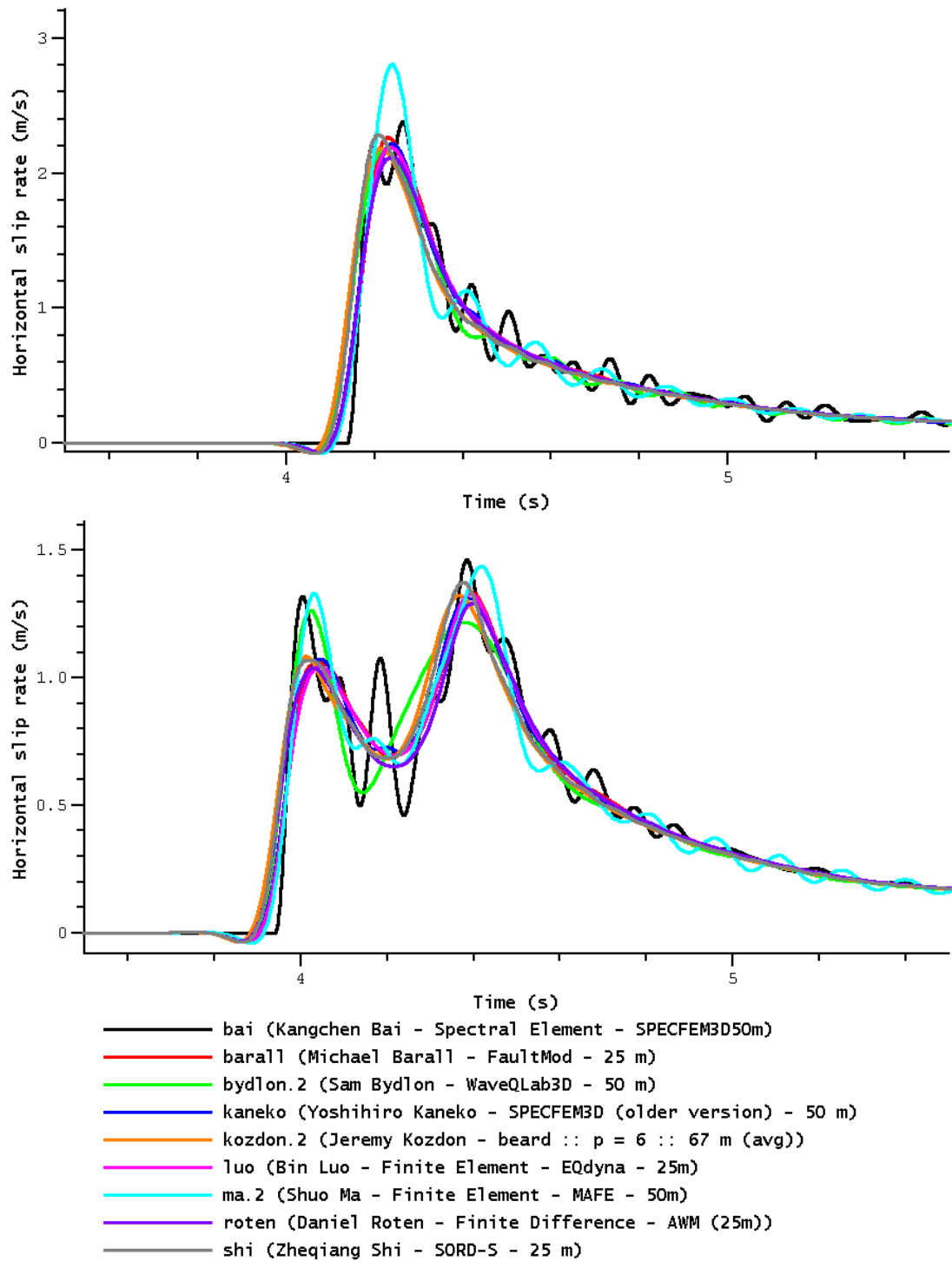
Figure 18. TPV32 waveforms for slip rate, at the epicenter (top graph), and at a station 200 m below the epicenter (bottom graph). It is challenging to get good results at these stations, as indicated by some disagreement among the code and some spurious oscillations. Note that the waveforms are quite different even though the stations are only 200 m apart.
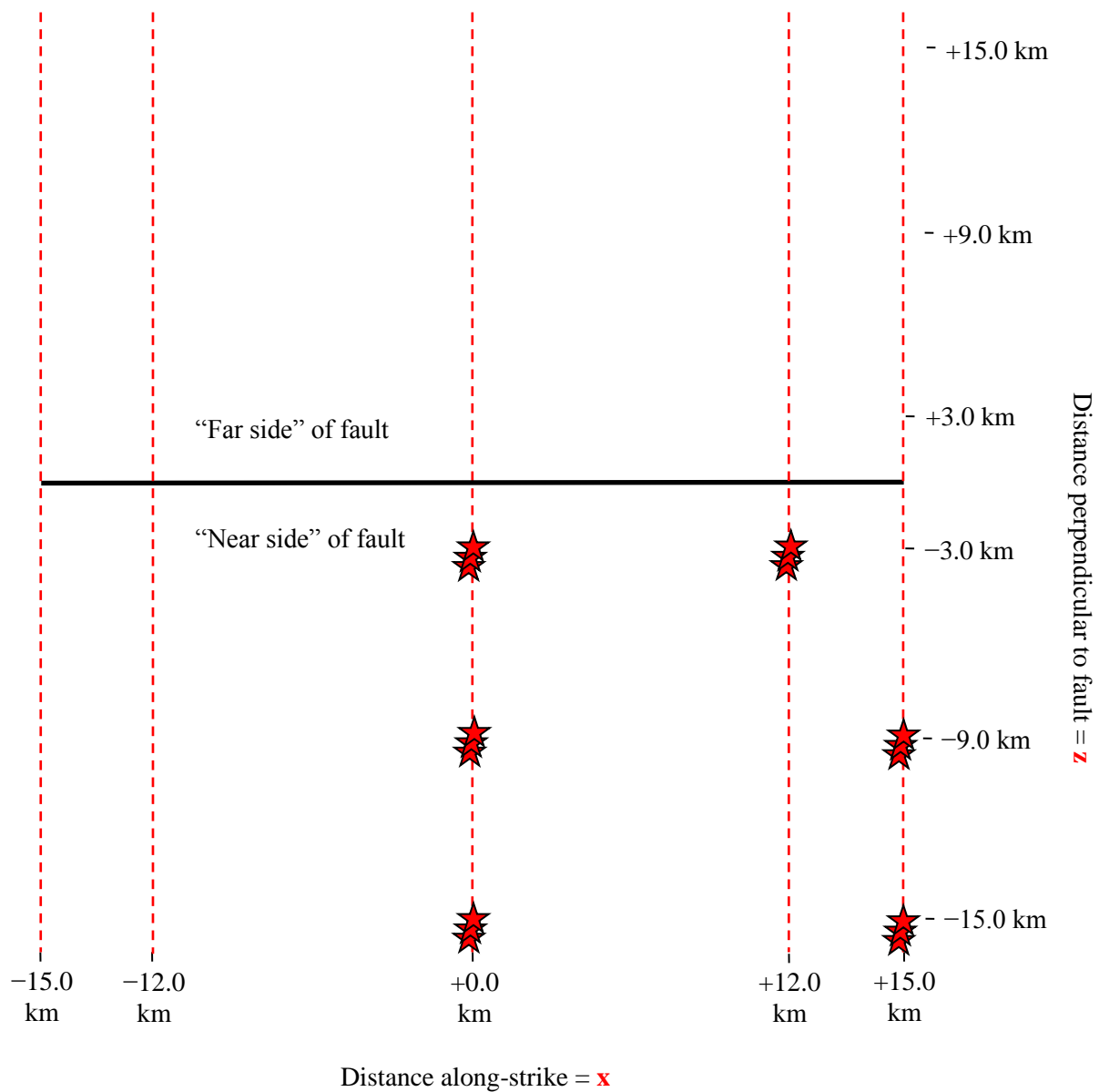
Figure 19. Off-fault stations for TPV31-32. The figure shows station locations on the earth's surface, in map view. This year, for the first time, we are placing off-fault stations far from the fault. Here, there are stations located 3 km, 9 km, and 15 km from the fault. Also new is that the stations are organized into 6 "boreholes" with stations at depths of 0.0, 0.5, and 2.4 km, to explore how the synthetic seismograms vary with depth in the 1D velocity structure.
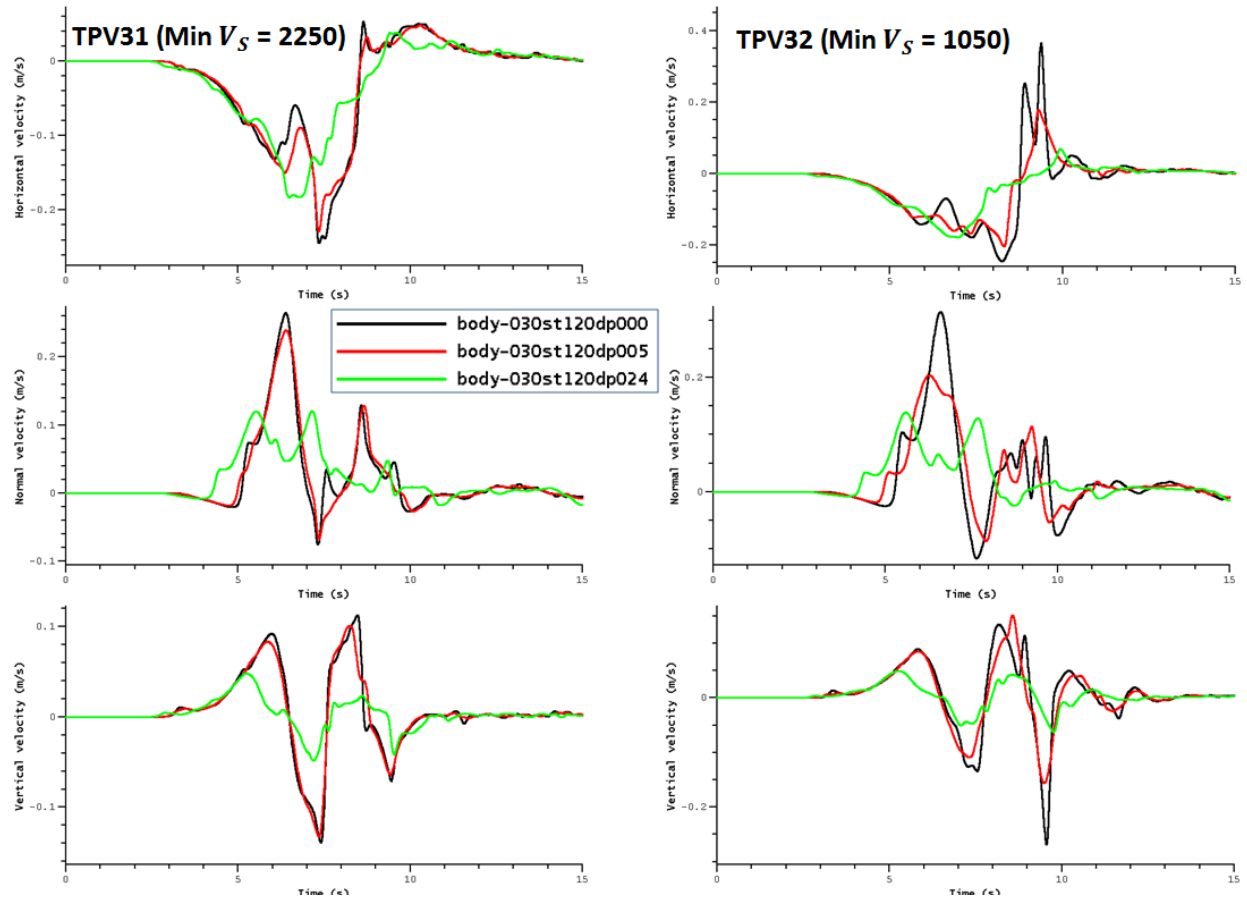
Figure 20. Synthetic seismograms for stations located 3 km from the fault. The left column is for TPV31, the right column for TPV32. The three rows are the horizontal (fault-parallel), normal, and vertical components of motion. Black curves are for the earth's surface, red for a depth of 0.5 km, and green for a depth of 2.4 km. For benchmark TPV31, the black and red curves are quite similar, but the green curves show less motion. For benchmark TPV32, the black curves have much higher peaks than the red curves, illustrating that the lower velocity in TPV32 amplifies the motion at the earth's surface.
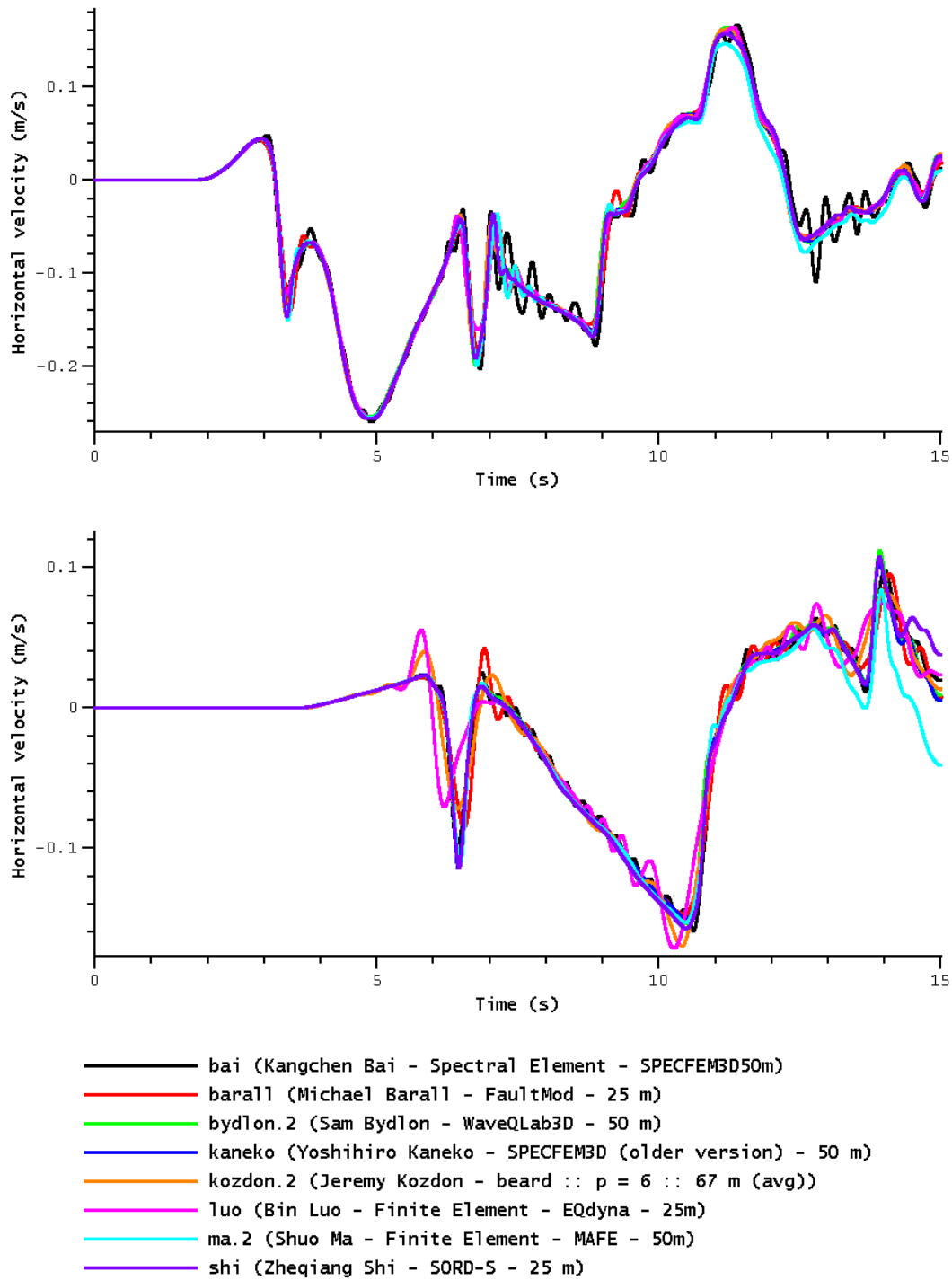
Figure 21. TPV32 synthetic seismograms at the earth's surface, for stations located 3 km from the fault (top graph) and 15 km from the fault (bottom graph). Due to the low velocity at the earth's surface, codes are in good but not perfect agreement at 3 km. Noticeable disagreements among the codes appear at 15 km.
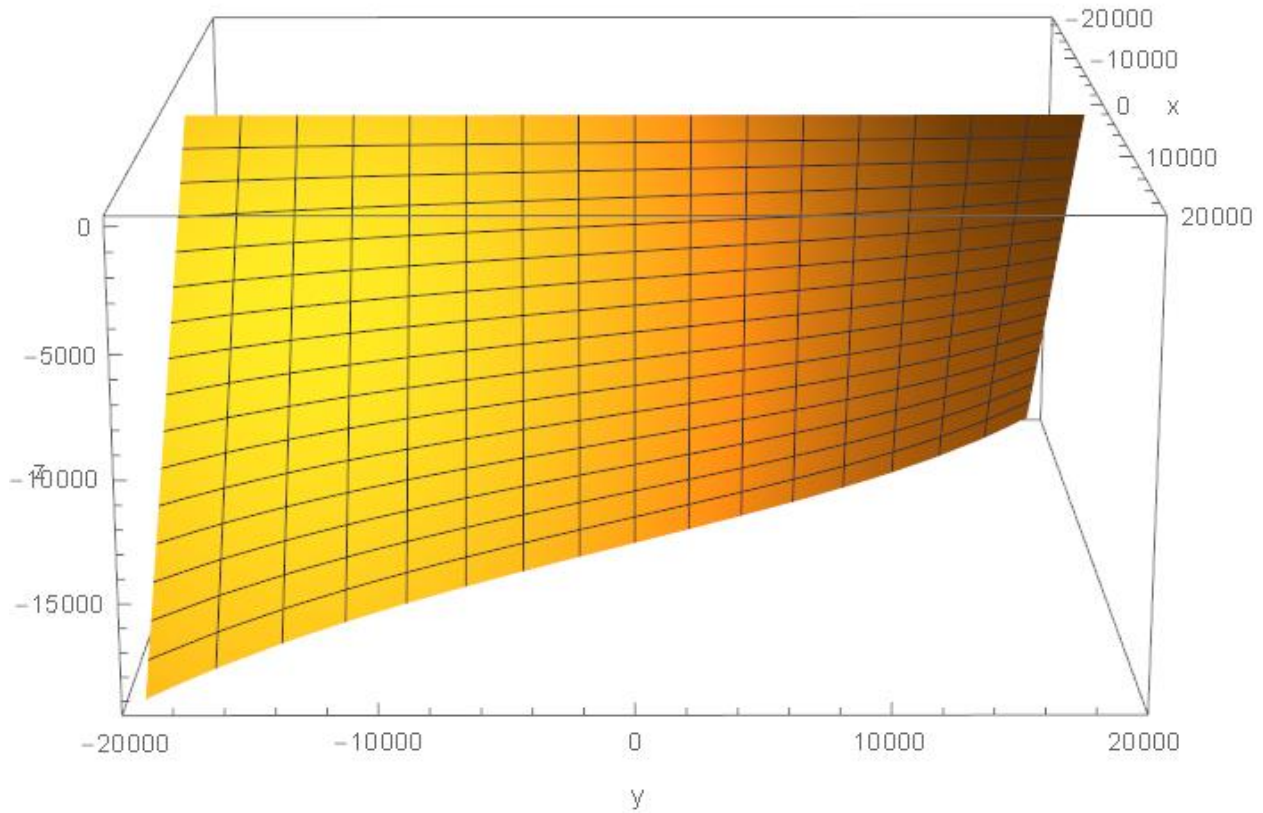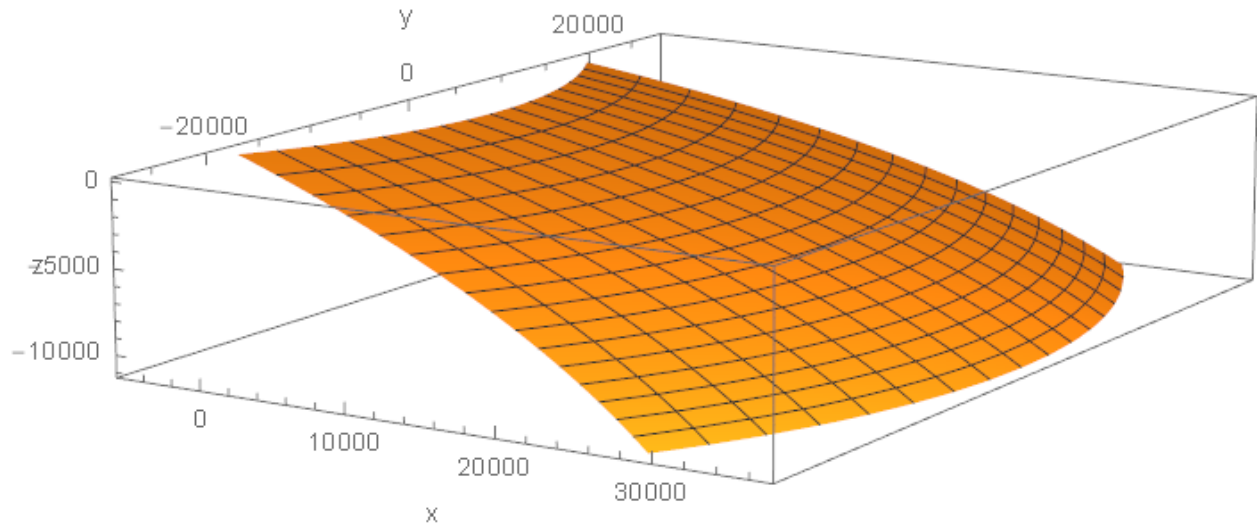
Figure 22. Negatively-curved fault surface, for testing the performance of triangular fault elements in earthquake simulators. The surface is a section of a helicoid. All dimensions are in meters. The fault trace is a straight line, at $x = 0$, extending from $y = -19200$ to $y = +19200$. Fault dip is −45 degrees at one end of the fault, and +45 degrees at the other end of the fault. Dip is 90 degrees in the center of the fault, at $y = 0$. The fault extends from the earth's surface at $z = 0$, to a maximum depth of $z = -19200$. This is a strike-slip fault. Grid lines are contours of constant depth (z) and contours of constant distance-along-strike (d), and do not represent fault elements.

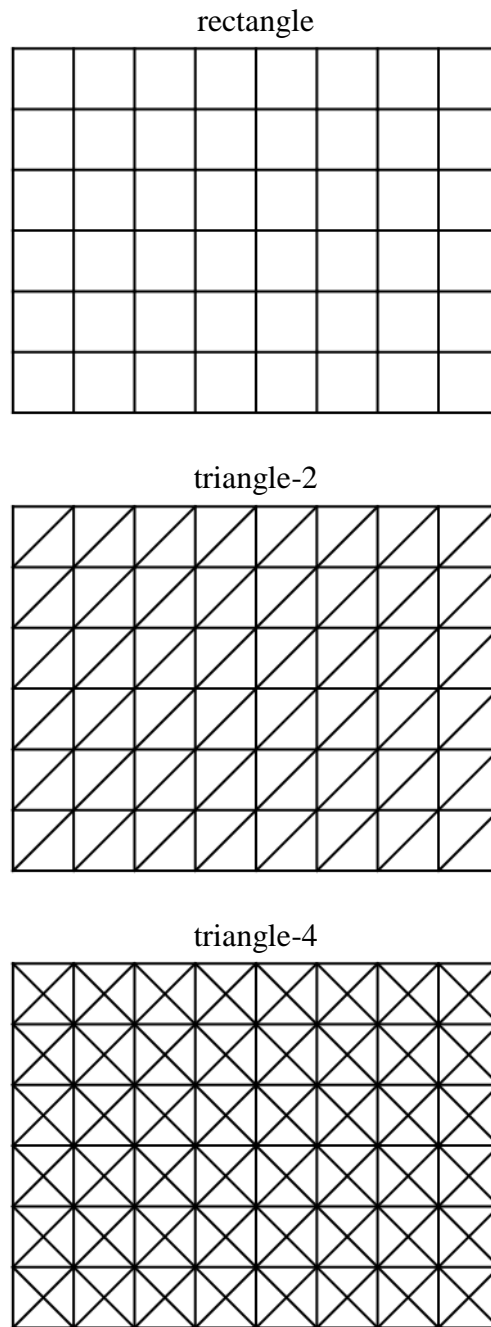**Figure 23.** Positively-curved fault surface, for testing the performance of triangular fault elements in earthquake simulators. The surface is a section of an ellipsoid. All dimensions are in meters. The fault trace is a curve, whose strike angle varies from −30 degrees to +30 degrees. Fault dip is 10 degrees at the top center of the fault, and 30 degrees at the bottom center of the fault. The fault extends from the earth's surface at $z = 0$, to a maximum depth of approximately $z = -11065$. This is a dip-slip (thrust) fault. Grid lines are contours of constant depth (z) and contours of constant distance-along-strike (d), and do not represent fault elements.

31

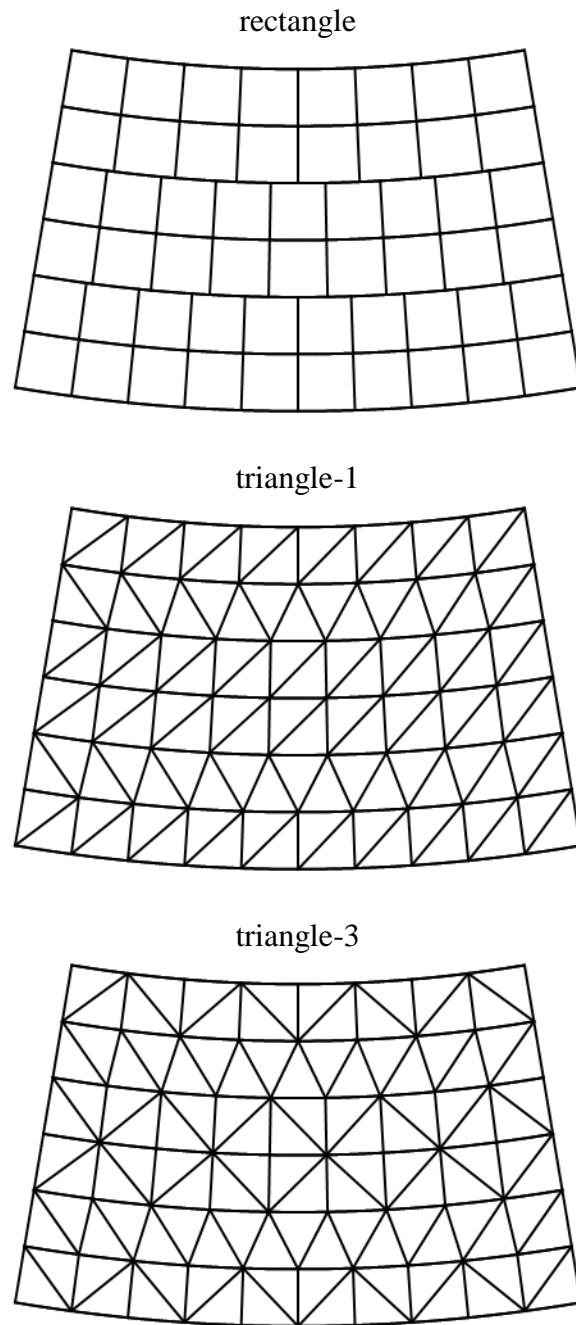rectangle

triangle-2

triangle-4

Figure 24. Discretization patterns for the negatively-curved (helicoidal) fault surface. The figure shows three different ways that a portion of the fault surface is partitioned into fault elements. The three methods are "rectangle", "triangle-2", and "triangle-4". For the "rectangle" method, the figure shows the fault elements as non-planar quadrilaterals, before they are converted into perfect rectangles. Conversion into perfect rectangles will create gaps and overlaps between adjacent rectangular fault elements. The figure is not to scale.

rectangle

triangle-1

triangle-3

Figure 25. Discretization patterns for the positively-curved (ellipsoidal) fault surface. The figure shows three different ways that a portion of the fault surface is partitioned into fault elements. The three methods are "rectangle", "triangle-1", and "triangle-3". For the "rectangle" method, the figure shows the fault elements as non-planar quadrilaterals, before they are converted into perfect rectangles. Conversion into perfect rectangles will create gaps and overlaps between adjacent rectangular fault elements. Note that unlike on our negatively-curved surface, the "rectangle" method does not produce a checkerboard pattern. The figure is not to scale.
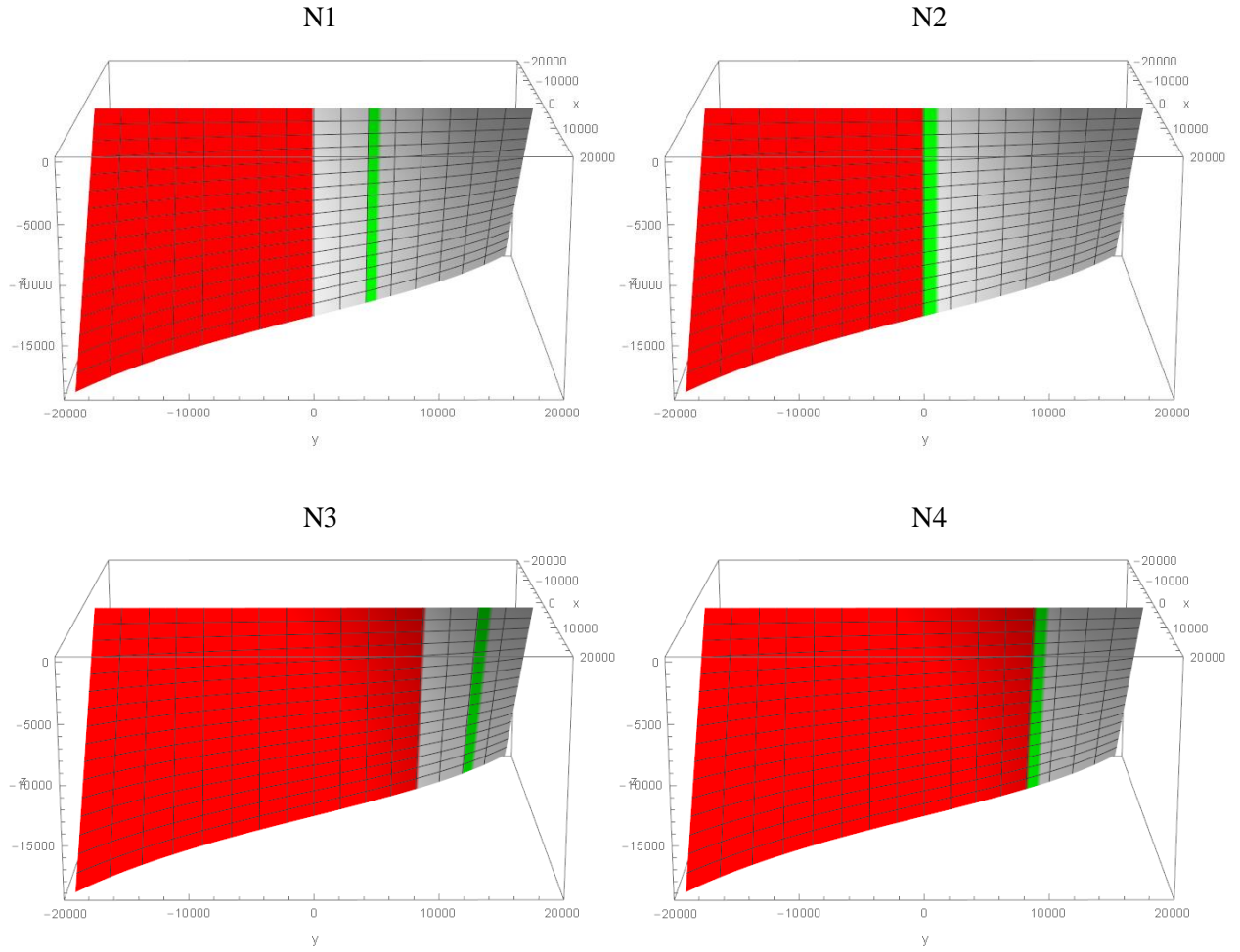
Figure 26. Source and target configurations for the negatively-curved (helicoidal) fault surface. The four configurations are designated N1 through N4. The source region is shown in red, and the target region is shown in green. The target region is a strip one element thick; the figures assume an element size of 1200 m. Grid lines are contours of constant depth (z) and contours of constant distance-along-strike (d), and do not represent fault elements.
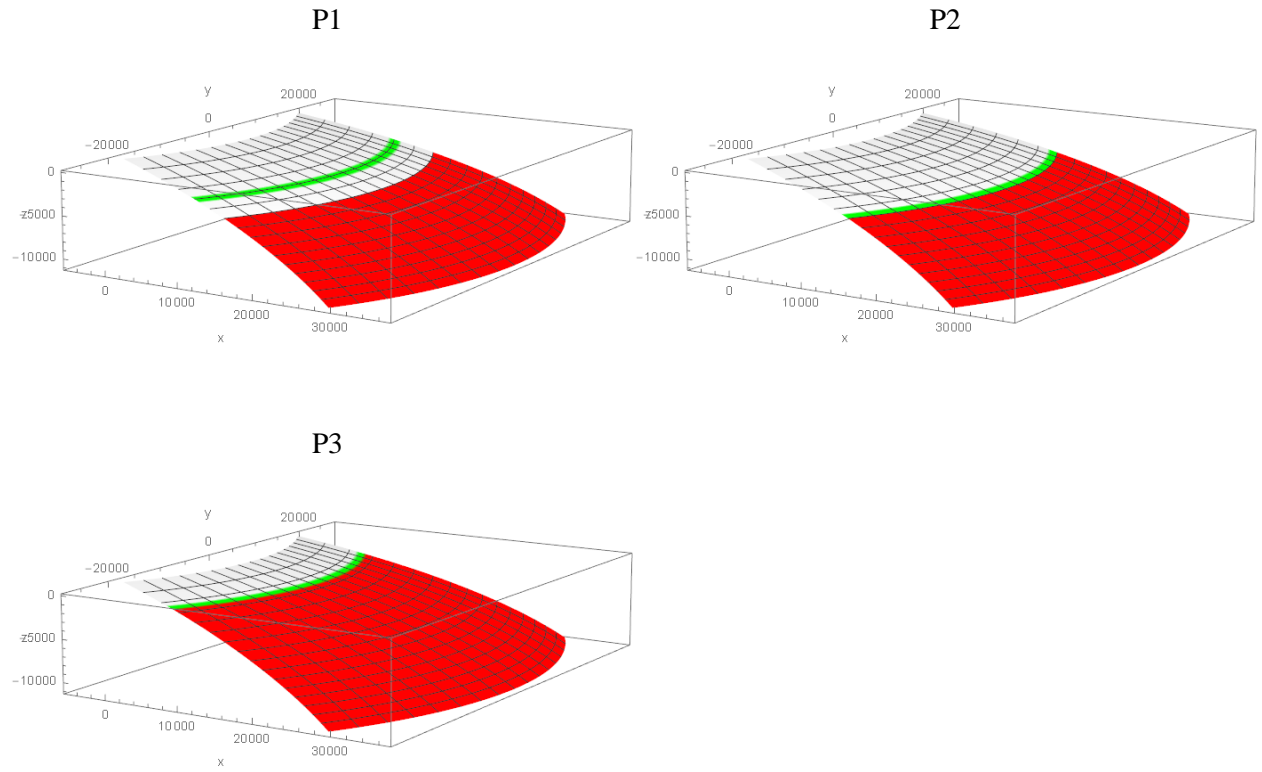
P1

P2

P3

Figure 27. Source and target configurations for the positively-curved (ellipsoidal) fault surface. The three configurations are designated P1 through P3. The source region is shown in red, and the target region is shown in green. The target region is a strip one element thick; the figures assume an element size of 1200 m. Grid lines are contours of constant depth (z) and contours of constant distance-along-strike (d), and do not represent fault elements.

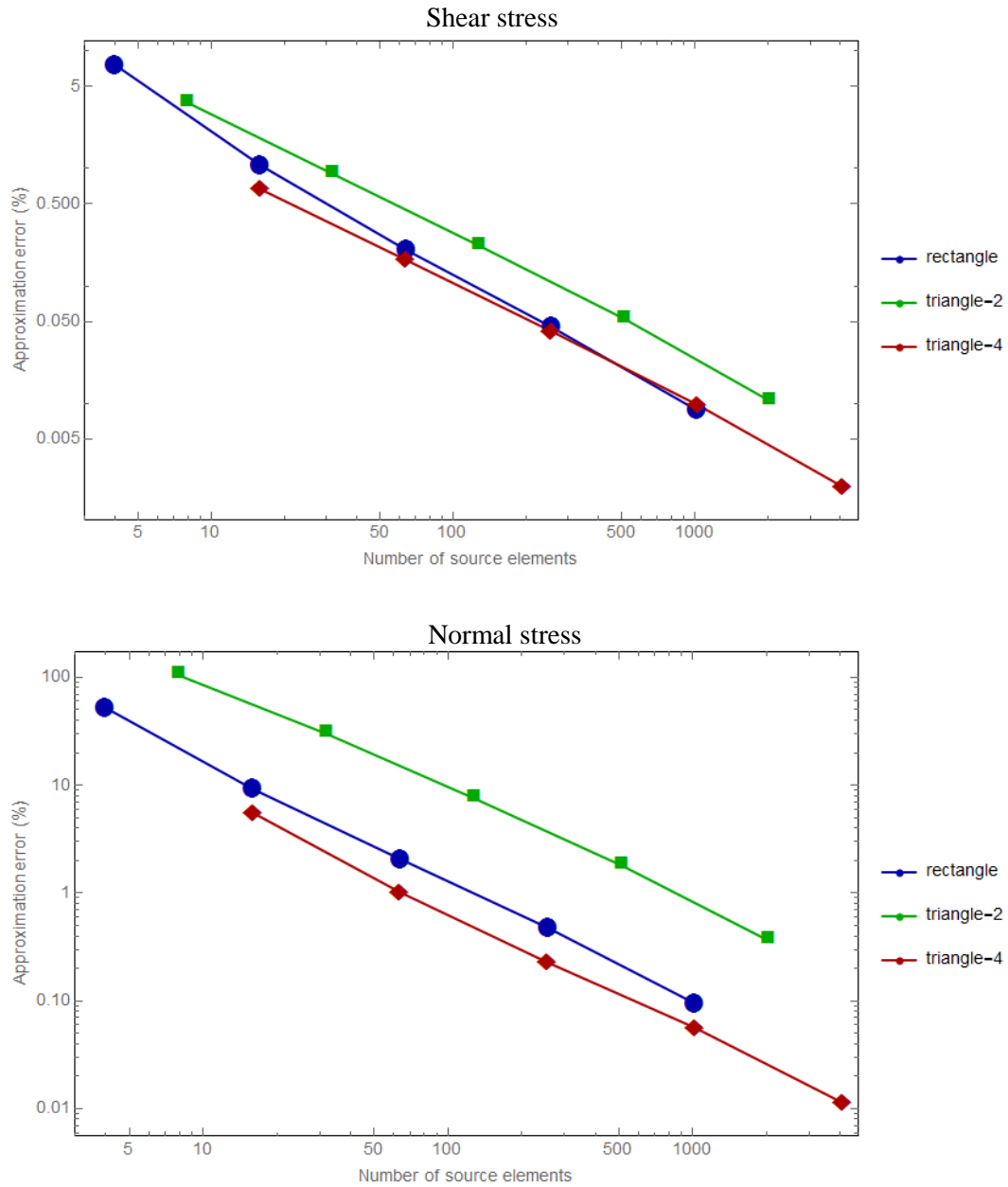**Source Test: Negatively Curved Fault Configuration N1**

Figure 28. Source test on the negatively-curved (helicoidal) fault surface. The vertical axis is approximation error, in percent; lower values are better. For shear stress, rectangle and triangle-4 perform about equally well, and both are better than triangle-2. For normal stress, triangle-4 is a little better than rectangle, which in turn is much better than triangle-2.

Target Test: Positively Curved Fault Configuration P1
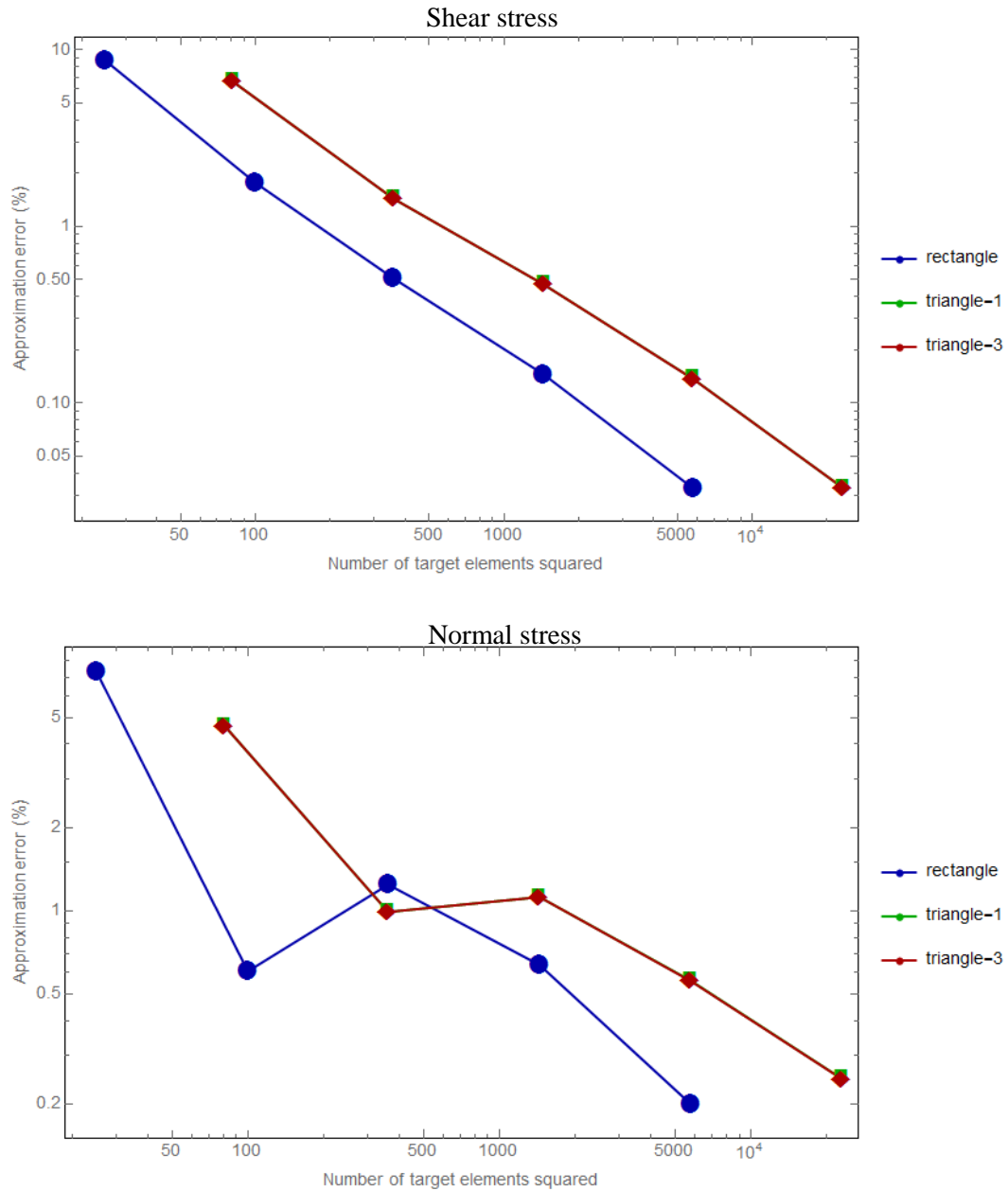

Figure 29. Target test on the positively-curved (ellipsoidal) fault surface. The vertical axis is approximation error, in percent; lower values are better. Rectangle out-performs both triangle-1 and triangle-3, for both shear and normal stress, except for one anomalous point on the normal stress plot.

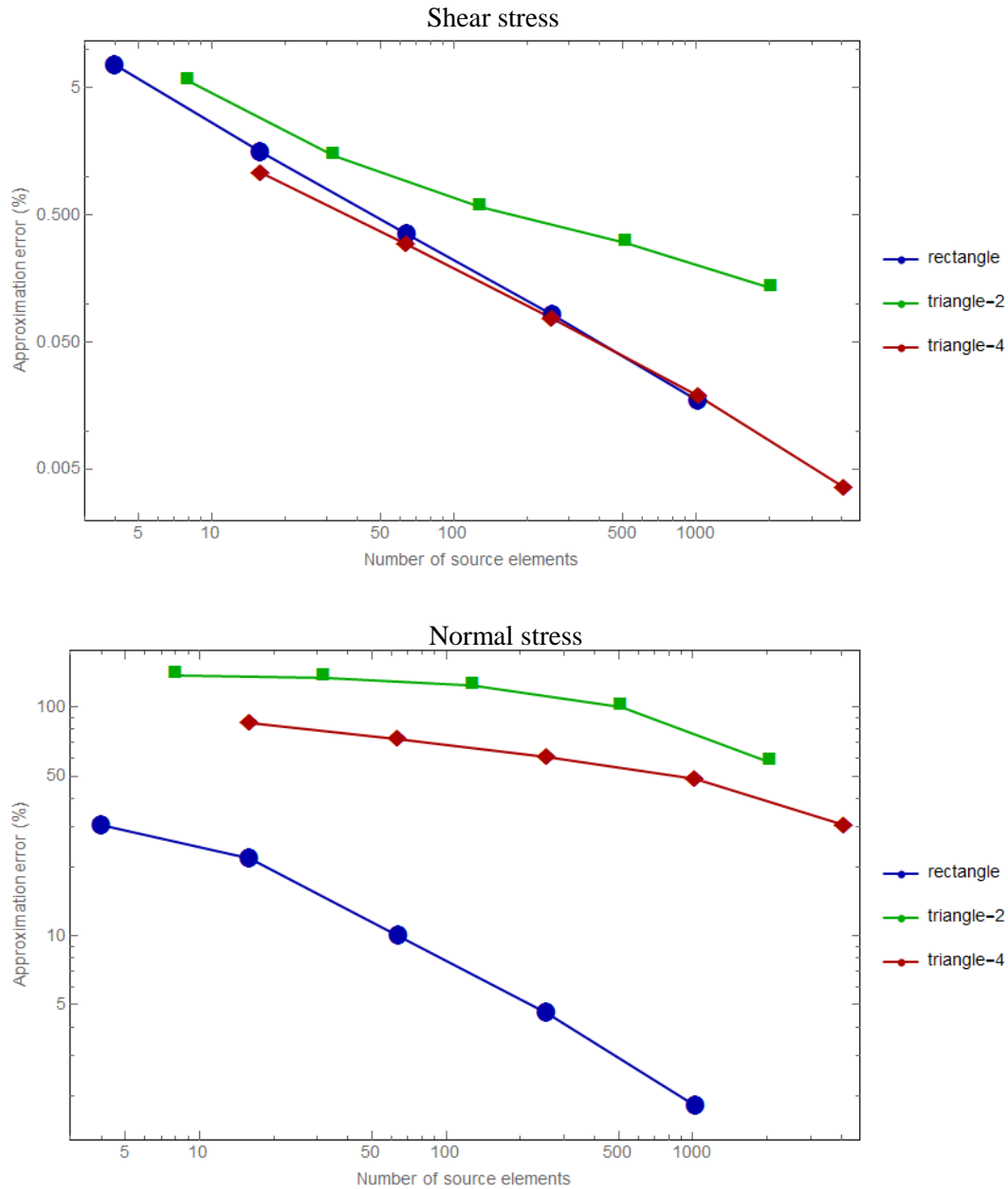**Propagation Test: Negatively Curved Fault Configuration N2**

Figure 30. Propagation test on the negatively-curved (helicoidal) fault surface. The vertical axis is approximation error, in percent; lower values are better. For shear stress, rectangle and triangle-4 perform about equally well, and both are better than triangle-2. For normal stress, rectangle performs much better than triangle-2 or triangle-4. We attribute the large errors in normal stress for triangular elements to the "corrugation" that occurs when triangulating the negatively-curved surface. Bear in mind that in the propagation test, the target elements are almost co-planar with the adjacent source elements, so that small variations in strike and dip angles can produce large percentage errors in the normal stress.

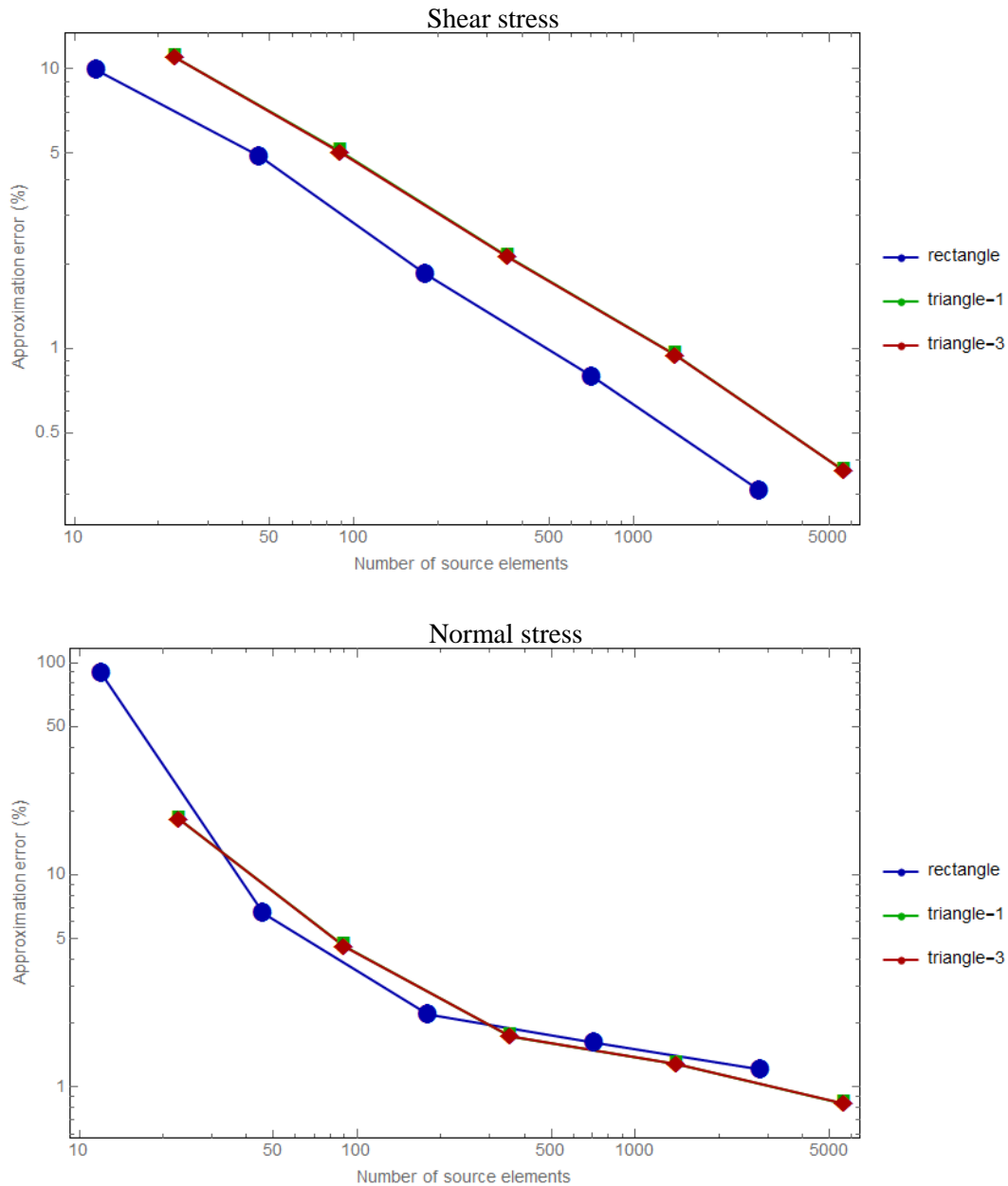**Propagation Test: Positively Curved Fault Configuration P2**

Figure 31. Propagation test on the positively-curved (ellipsoidal) fault surface. The vertical axis is approximation error, in percent; lower values are better. For shear stress, rectangle out-performs both triangle-1 and triangle-3. For normal stress, rectangle, triangle-1, and triangle-3 all perform about equally well. The accuracy seems to be leveling off at about 1 percent. One possible explanation for why accuracy might level off is that as elements get smaller, they follow the surface more accurately (which tends to decrease error), but also their centroids are closer together (which tends to increase error). The normal stress results are very different than for the negatively-curved surface, where the triangular elements performed very poorly.